computing today

ISSN 0142-7210
JUN 1979
50p

COMPUTER GAMES SURVEY

THE LIGHTER SIDE OF THE HOBBY!

COMPUCOLOUR REVIEW

BUS ROUTE FOR S100

TRITON HUMBUG

computing today

VOL 1.NO 4.
JUNE '79

Are you game? We'll play along!

PAGE

Distributed by Argus Distribution Ltd. Printed by LSG. Limited, Lincoln.

EDITORIAL AND ADVERTISEMENT OFFICE
25-27 Oxford Street, London W1R 1RF. Telephone 01-434-1781/2 Telex 8811896

CLUB NEWS

It is proposed to set up a club in Britain for those people using the RCA 1802 microprocessor, Cosmac ELF, ELFII, Super Elf etc. The unofficial assistance of RCA and HL Audio has been promised. Would those interested please contact James Cunningham at 7 Harrowden Court, Harrowden Road, Luton LU2 0SR. Enclose an SAE please.

A computer club has been set up at Southgate Technical College and anyone requiring more information should write to Mr I.E.Williams at the college, High Street, London N14 6BS.

A new computer club has been formed in South Yorkshire. Known as SYPCG it has been created for those who are primarily interested in computing on a do-it-yourself basis. Meetings will be held on the 2nd Wednesday of each month and it is hoped that profiles of equipment and problem solving sessions will be featured. bership is £3 for 1979 and all people in the area are welcomed to join. The address for further information is :- Tony Rycroft, SUPCG Secretary, 88 Spinneyfield, Moorgate, Rotherham, S.Yorks. Meetings will be held at the University of Sheffield.

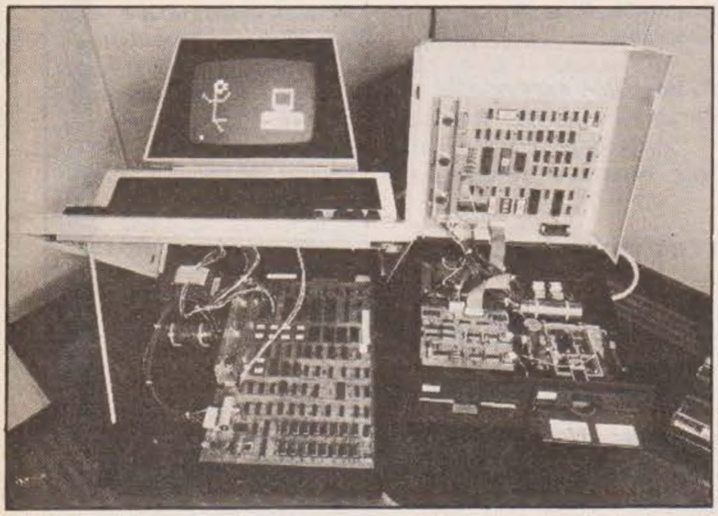
Another addition to the computer club scene is SPEC, the Sorcerer Program Exchange Club. Formed to promote the Exidy Sorcerer the club will act as a clearing house for program ideas and helpful hints on the use of the machine. The news letter and further information is available from Mr. M.P. Hannaby at 65 Trafalgar Road, Birkdale, Southport PR8 2NJ.

BUSINESS SYSTEMS LAUNCH

Commodore have announced an endorsement scheme for PET compatible products. The first to be included in the scheme is a new range of business software from ACT, Petsoft's parent company, under the name PETACT. Launched as a total system the hardware sells at under £2,500 which is less than half of commercially produced systems. The software prices range from £225 to the complete system at around £800. You can select either disk or cassette based softwaree and there is no problem upgrading from one to the other because the programs are written in BASIC. The hardware consists of the new 32K large keyboard PET, a twin floppy drive and a tractor feed printer. All the Commodore products and the first is currently being delivered and the second two will arrive in your local shop about mid May. The software is the first set of business software to be written for PET by professional company and appears very impressive, the first two packages announced are Purchase and Sales accounting. Chuck Peddle, the father of both KIM and PET, was present and hinted that there are some very interesting hardware products on the way in the coming year. Sales figures of the PET in the UK are very interesting, during 1978 around 3000 were sold in the country and already this year another 3000 have been ordered, about 70% are going to the professional market. The UK dealer network is running at about 100 at the moment and these have all been selected on the basis of being able to provide both hardware and software support for the customers. On the subject of the new PETs with expanded memory and large keys they look most impressive but the proposed new green display has not yet been implemented, the large keys are well labelled with block letters, possibly too large, and are double injected to avoid the wear problem that happened with the earlier types. The graphics legend is engraved on the leading edge of the key face. Although there is no Commodore memory expansion yet it is hoped to approve some under the endorsement scheme so you will be able to upgrade your system to cater with the floppy disks. A final note on the business system is that the price of software includes a days training for an operator in the use of it.

MOTOROLA ROLLS ON

Motorola have just announced a vast array of new chips and micro support devices. The new chips are the MC6805 8 bit CPU containing clock, 64 bytes RAM, 1100 bytes ROM, memory mapped I/O and timer, the MC6882 eight bit buffer latch and two memory chips. There are the TMS2716 EPROM in 2024 by 8 format with industry standard pin-out and the new 4K by one static RAM. This also has the standard pin-out and operates from a single 5 V supply. The support devices include a new dual drive, double sided floppy for the EXORcisor, EXORterm, and Micromodule products. Also included in the release are a new CRT terminal for use with the EXORciser and a development package for use with the 6801 micro. More information on all these new products can be obtained from Motorola at York House, Empire Way, Wembley, Middx.

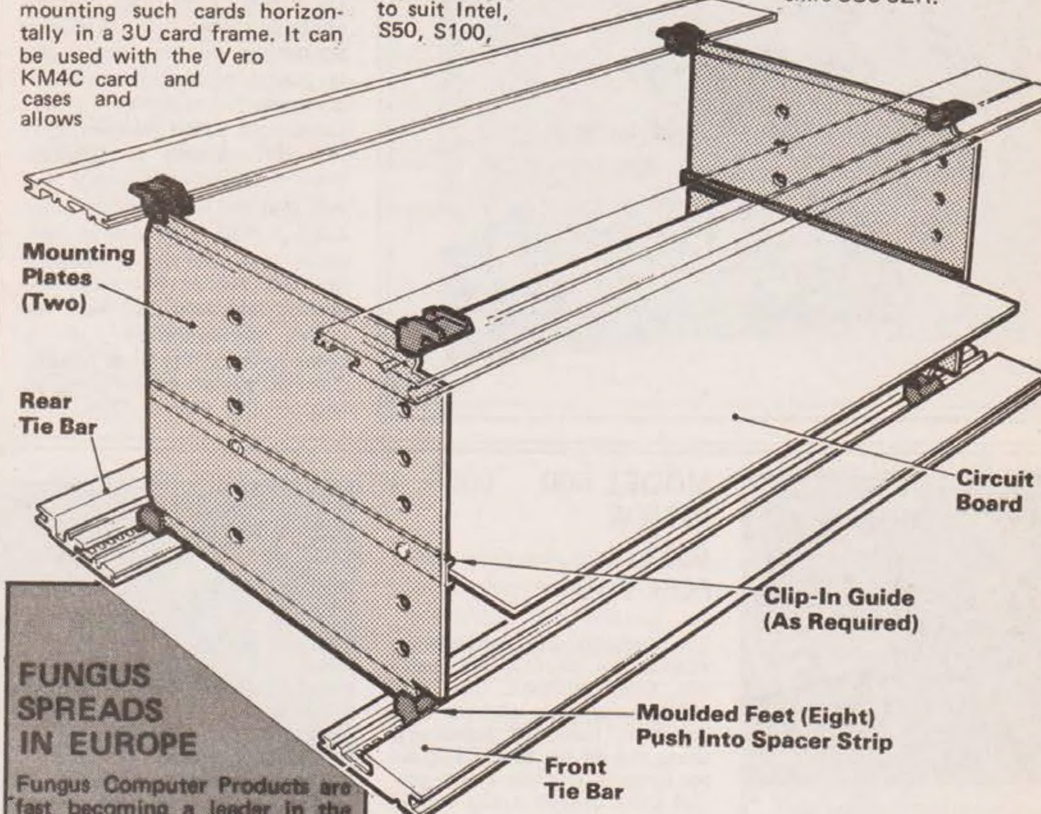


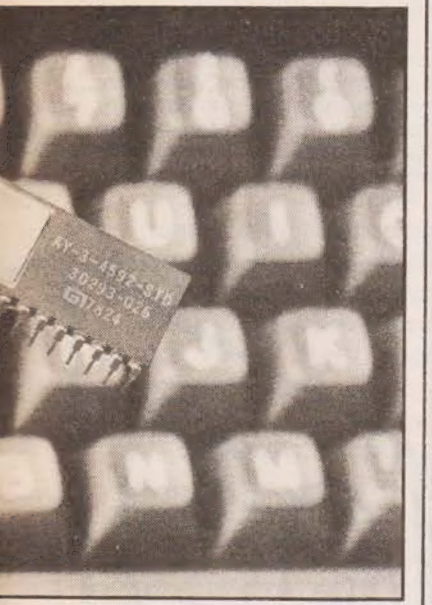
AIM 65 NEW VERSIONS

Portable Microsystems of Brackley, Northants have the Rockwell AIM 65 micro in three new versions. These are the standard board system at £249.50, a cased version complete with power supply, 4K RAM and 8K ROM BASIC. The unit is mounted in a desk-top case and is designated the 56C. Selling price is from £485. The third version is the PDS 65 and is a development system. Produced for programmers and engineers it allows modification and de-bugging work to be undertaken on a customers system. The addition of an optional acoustic coupler allows the field engineer to access the host computer from the point of work. Prices start at £950. For more information contact PM on Brackley 702017.

VERO--RACK FOR S100/KM4C

Vero Electronics have launched a new card frame system to solve the designers problem with large format cards. They have introduced a kit for mounting such cards horizontally in a 3U card frame. It can be used with the Vero KM4C card and cases and allows up to six in a single frame. Supplied with two mounting plates which can be positioned anywhere within the frame to suit Intel, S50, S100, Motorola, or Double Eurocard sizes. For more information contact Vero at Industrial Estate, Chandler's Ford, Hampshire SO5 3ZR.





FUNGUS SPREADS IN EUROPE

Fungus Computer Products are fast becoming a leader in the field of compatibles in the UK and Western Europe. The range is being extended to include PDP 11 compatible products, LSI 11 compatible products, PDP 11/70 memories, Large storage module disk sub-systems and the highly successful range of products previously marketed by BML. For further information contact:- Pam Ryan, No. 1 Westmoreland House, 2nd Floor - Teall Street, Wakefield, West Yorkshire.

D2 INTERFACE UNIT

One of the only problems with the Motorola D2 evaluation kit is that it cannot easily talk to the outside world. Now Mektronic, a Manchester based company, have announced an interfacing unit for the D2 which changes that. Selling at an inclusive price of £158.76 the unit is ready built and supplied with software listings. The unit connects to the spare PIA on the D2 and gives 16 input or output channels. Powered by the D2's supply the unit can direct drive devices using 0—24 V at 150 mA. For further details contact Mektronic at Linden House, 116 Rectory Lane, Prestwick, Manchester M25 5DB.

KEYBOARD CHIP

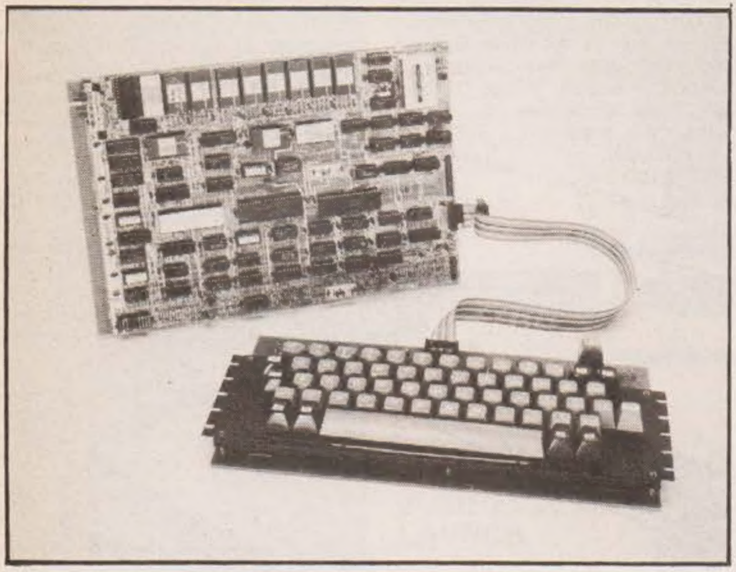
A keyboard encoder IC that can cater for all types of signals, capacitive, Hall effect or switch closure, has been produced by General Instrument Microelectronics. Using NMOS circuitry the 40 pin chiip uses pulse detection techniques to cater for 128 keys and has internal protection to deal with key bounce and noise. The keys are connected in a 16 by 8 matrix of which 112 can have up to four 10 bit programmable codes. The remaining 16 keys are eserved for discreet functions. The internal control oscillator scans the keyboard in 1.7 mS which allows burst typing of up to 250 wpm. Power requirements are 5 V and all outputs are TTL and CMOS compatible.

SWIFT SASCO PROGRAM BY STAG

Announced this month by Swift Sasco is a programming service for PROMS using a STAG Model PPX universal programmer. It can cope with all the currently available range of Signetics and National PROMS and EPROMS. It is hoped to give a 24 hour turn round service for prototypes and larger quantities can be arranged for the customer. The program can be supplied in one of three forms — truth table, punched tape or a master PROM. Swift Sasco will supply the full range of PROMS for customers own programming. The STAG allows for programming of the new PLA's and PGA's recently announced by Signetics. Contact Swift Sasco at PO Box 2000, Crawley, Sussex.

POWER FROM HAL

Power-One, the largest producer of open frame PSU's have appointed HAL Computers of Weybridge as sole UK distributors. They offer 80 off-the-shelf supplies including ones suitable for the industry standard Shugart mini-floppy drives. All DC outputs are overvoltage protected as standard, not usually found. This is the first venture into the personal computer market by HAL and it is expected that more products will be announced soon. Contact HAL at 133 Woodham Lane, New Haw, Weybridge, Surrey.



NASCOM (SUPER) 2 IN JUNE

On Friday, 20th April Nascom Microcomputers Limited; announced to the world the launching of their latest small computer system; not suprisingly called Nascom 2. It will be available in quantity from the summer onwards and will cost the lucky owners only £295 + that Very Awful Tax. Until a system becomes available for us to review we cannot comment but here for your consideration is a breakdown of the press release. The new system will be designed around the 4MHz Z80A CPU which increases the running speed over the NASCOM 1 by about 1¾ times. It comes with a quality PCB and full supporting cast of chips including a 16-bit programmable Input/Output port, a much improved cassette tape interface using the Kansas City (CUTS) format at 1200 or 300 BAUD, and an uprated keyboard. The control is provided by a 2K monitor known as NAS-SYS 1. This is completely new and offers 22 commands for entering, modifying and displaying programs, directly accessing input and output port, and making intelligent copies.

The board will contain one chip MK36000 which is a 64K bit Rom and this will hold the 8K Basic, which is an adaption of the Microsoft Basic used by many notable companies. With this

AMPLICON POWER UP

Amplicon are supplying Stevens-Arnold series PSU's ex stock. These deliver +5 V @ 3 A, +12 @ 1 A, and -5 @ ½ A and are suitable for the 8080 and similar CPU's. Standard inputs are 12, 24, 28, 48 and 60 volts DC. The 1-9 price is £151. Contact Amplicon at Lion Mews, Hove BN3 5RA.

S100 VDU BOARDS

Many of you have written in to ask for the PCB foils for our S100 VDU project and as the response has been so good the actual boards will now be produced. Tamtronik are going to make it for us and you should contact them on 021-557 9144. Approximate cost will be £20.

MODEL 500 IS NEW 6502 FOR YOU

The Model 500 CPU card is being stocked by MUTEK of Quarry Hill, Box, Wiltshire. The card is fully compatible with the Ohio Scientific range of accessories using the 48 pin bus. Designed for systems engineers and general OEM useage it has a 6502 processor, an on-board 8K Microsoft BASIC with six digit floating point, 4K RAM and an RS232 or 20 mA serial interface. Options include a PIA, 256K memory management capability and up to three 1702 type ROM's. The available accessory boards include an audio cassette interface, graphics board, floppy disk controllers and the full range of peripherals. Power requirements are +5 and −9 volts. Contact Mutek on Box 3289.

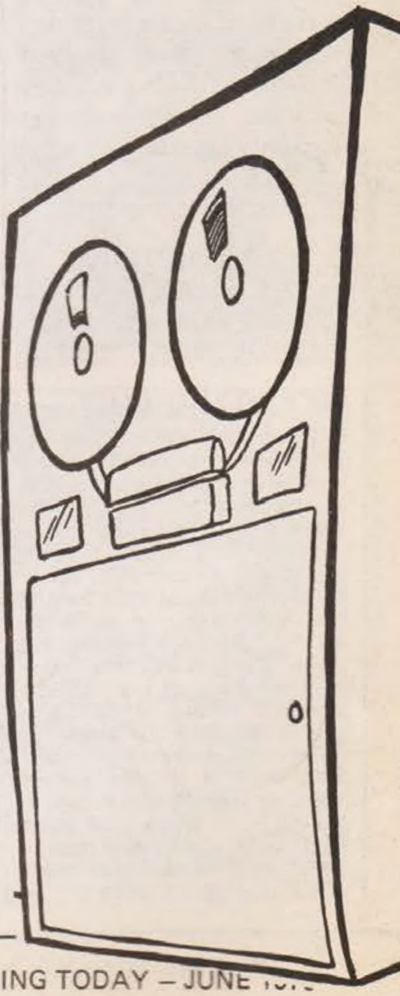
DIGITAL MEMORY REPLACEMENT

A single card replacement for four PDP 11/04 or 34 memory cards supplies the full 256K bytes (128 by 18) in one single backplane slot. Designated the NS11/34Q it has been announced by National Semiconductors and has a read access time of 300 mS which is twice as fast as the DEC version. The board includes the parity check and generation which would normally be performed by the M7850 controller. On-board DIP switches set the memory boundaries in increments of 4K. Power requirements are 2.4 A @ 5 V and 1 A @ 15 V. All devices are socketed for easy replacement. Further details from National on (0234) 211262.

MARK OF RESPECT

Apologies are due to Mark Strathern of Lotus Sound because we forgot to credit him for loaning us the Computhink unit from his stocks last month.

The unit shown on our cover belonged to him, and he also provided invaluable assistance in the preparation of the article. Sorry Mark!



VDU FOR YOU

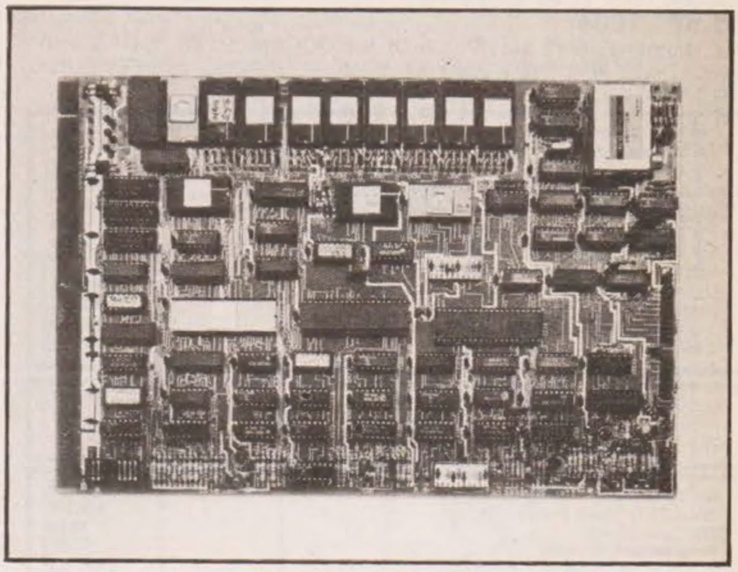
A new VDU for the low-cost market has been announced by Applied Digital Display Systems and is being marketed by Terminal Display Systems under the name Regent 20. The device can work in half or full duplex modes and uses a five by eight matrix on a 24 by 80 screen format. Included as standard are a printer interfsce, line monitor, RS232 or 200 mA serial interface, cursor options and a switch selectable character set. The unit is easily accessed for servicing and has an expected MTBF of 7000 hours. For more information contact TDS on 0254 662244.

BASIC it should be possible to run most of the basic listings published today without too much modification. (Don't worry NASCOM 1 owners, we will still publish machine code listings — Ed).

An unusual set of commands is the DEEK and DOKE. These are 16 bit versions of PEEK and POKE.

Altogether the board holds 20K of addressable RAM, and the rest of the space not mentioned is for a cleaned-up memory mapped display (1K), a 1K scratchpad used by both the monitor and the user, and 8K of user ram which can be expanded (off board) to 32K. The equipment practice is identical to that of the NASCOM 1 with a good industrial PCB which will plug into the motherboard already marketed. One difference to the '1' is that all address lines are buffered on board, although if this renders the existing buffer-board redundant I'm not quite sure.

Altogether a system with 8K of user RAM, 8K monitor and 4MHz operation is very impressive. When this is looked at with the other features common to the NASCOM 1 then it certainly warrants a greater inspection. Here in these few lines I have not tried to document all the pros & cons I leave that to the distributors but despite the Managing Directors assertions that this is not a replacement for the NASCOM 1, I feel that for the extra £130 prospective buyers are going to have a lot of heart searching and arguments with the wife.



CORRECTION TO NASCOM TUNES

A slight musical discrepancy occurred in piece "Fuer Elise" in last month's issue. Line 24 of this should read:-

5A F3 58 F2 5C F3 58 51

The program is bug-free!

SECOND GENERATION 6800

The evergreen 6800 micro has spawned a second generation version designated the 6802. Retaining all the features of the 6800 it has the added features of an on-chip clock and 128 bytes of RAM. This reduces component count in support circuits, the bulky 1MHz crystal is replaced by a 4MHz one and the first 32 bytes of the on-chip RAM can be held in low power mode by a Vcc standby during power failure. Expandible up to 65K it is available in plastic or ceramic 40 pin DIP's. Fo more information contact Jermyn on 0732-50144.

EDITORIAL ENQUIRIES

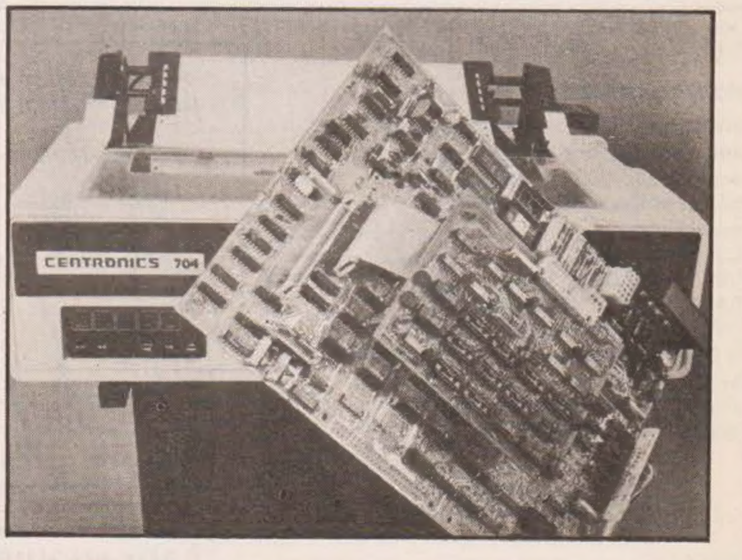
We regret that our editorial staff are unable to answer queries relating to the relative merits of the many personal computer systems on the market. However if you have any genuine queries regarding articles which have appeared in CT — including system reviews — write to us here, and enclose an SAE.

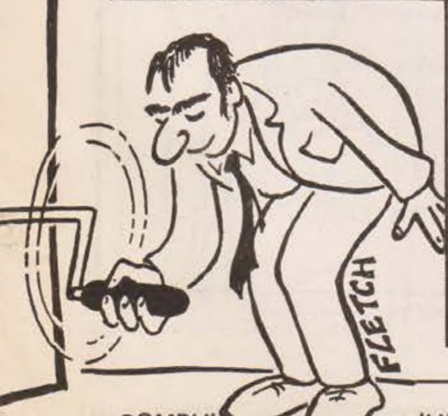
MICRODIGITAL CATALOGUE

A new brochure detailing the range of product stocked has been produced by Microdigital of Liverpool. The brochure includes details of the Mk 14, Newbear products, Acorn, AIM 65, Nascom, Apple 2 and the Exidy Sorcerer. Also included are more details on the hire service that we mentioned in last months news. All the prices for basic systems and available expansion are quoted but unfortunately the book list, Microdigital are stocking a vast selection, was not included. Further information on 051-236 0707.

NEW CENTRONICS PRINTERS

Two new printers are to be launched by the well known firm of Centronics. These are Models 753 and 704. The 753 is a new style of printer, aimed at the word processor market it will offer high quality output with a high document throughput. Micro controlled it features high density printing, proportional spacing, 130 tp 150 cps and a fast slew rate giving it a three times printing speed over the daisy wheel type. Using an N by 9 matrix head it can give fully formed characters with twice the density of a normal matrix head. Four configurations are available to suit your requirements. The Model 704 is aimed at the data communications market and has a fast print rate of 180 cps, giving up to 400 lines per minute. Using the well proven free flight head it can cater for a variety of paper thicknessess and will print the standard ASCII 96 character set. Baud rates of up to 9600 can be achieved through an RS232/V24 serial interface. More detail from Centronic on 01-581 1011.





NEW MICRO CAT

Rapid Recall have published a full price list for their stocks of micro's, memory and support IC's. It includes details of DEC, Intel and Intersil products and is arranged in types rather than manufacturer to make it easy to use. Priced at £1 it can be obtained from Rapid Recall at 6 Soho Mills, Wooburn Green, Bucks.

L.P. ENTERPRISES

Room CT
313 Kingston Road, Ilford
Essex, IG1 1PJ, England
01-553 1001

A COMPREHENSIVE SELECTION OF MICROCOMPUTER BOOKS AND MAGAZINES FOR THE HOBBYIST, EDUCATIONALIST, PROFESSIONAL AND RETAILER

Introduction to Microcomputers:	
Vol 0: Beginners Book	£5.95
Vol 1: Basic Concepts	£6.30
Vol 2: Some Real Microprocessors (without binder)	£18.95
Vol 2: Some Real Microprocessors (with binder)	£24.70
Vol 3: Some Real Support Devices (without binder)	£11.95
Vol 3: Some Real Support Devices (with binder)	£17.70
Updating subscription (6 issues) for Vol 2	£18.95
Updating subscription (6 issues) for Vol 3	£18.95
Updating subscriptions for Vol 2 & 3	£30.00
1 Updating issue (specify for Vol 2 or 3)	£4.00
1 Binder (Specify for Vol 2 or 3)	£5.75

6800 Programming for Logic Design	£6.30
8080 Programming for Logic Design	£6.30
Z80 Programming for Logic Design	£6.30

BASIC Computer Games	£5.50
What To Do After You Hit Return	£8.95
8080 Galaxy Game	£8.95
Programming the 6502	£7.95
Computer Rage (A Board Game)	£6.95
Artist and Computer	£3.95
PIMS — A data base management system	£5.95
Games, Tricks & Puzzles for a Hand Calculator	£2.49

Z80 Instruction Handbook	£2.95
8080 Programmers Pocket Guide	£1.95
8080 Hex Code Card	£1.95
8080 Octal Code Card	£1.95

Dr Dobbs Journal Vol 1	£10.00
Best of BYTE	£8.95
Scelbi BYTE Primer	£7.95
Best of Creative Computing Vol 1	£6.95
Best of Creative Computing Vol 2	£6.95
Best of MICRO — 6502 journal	£5.50

8080A/8085 Assmebly Language Programming	£6.45
6800 Assembly Language Programming	£6.45
8080 Software Gourmet Guide and Cookbook	£6.95
6800 Software Gourmet Guide and Cookbook	£6.95
6800 Tracer — an aid to 6800 Program D-Bugging	£3.95
Program Design	£4.25

Magazine Subscriptions:	UK Price	Overseas Price
Subscriptions start with 3 weeks		
MICRO-6502 Journal (12 issues)	£14.50	£15.00
Personal Computing (12 issues)	£16.00	£17.00
Interface Age (12 issues)	£20.00	£21.00
Dr Dobbs Journal (10 issues)	£13.00	£13.50
Computer Music Journal (4 issues)	£10.50	£11.00
Recreational Computing	£8.00	£8.50
BYTE (12 issues)	£21.00	£21.00
Creative Computing (12 issues)	£16.00	£16.50
Kilobaud (12 issues)	£20.00	£21.00

Microprocessors from Chips to Systems	£7.95
Microprocessor Interfacing Techniques	£8.15
Z80 Microcomputer Handbook	£7.50
TV Typewriter Cookbook	£7.50
T.T.L. Cookbook	£7.50
CMOS Cookbook	£7.95
IC OP-AMP Cookbook	£9.50
RTL Cookbook	£4.25
IC Timer Cookbook	£7.50
The Cheap Video Cookbook	£4.30

Introduction to Personal and Business Computing	£4.95
Getting Involved with your Own Computing	£4.75
Your Home Computer	£5.95
How to Profit from Your Personal Computer	£5.50
Calculating with BASIC puting	£4.95
Hobby Computers are Here	£3.95
New Hobby Computers	£3.95
Understanding Microcomputers and Small Computer Systems	£6.50

Users Guide to North Star BASIC	£10.00
Instant BASIC	£7.50
Basic BASIC	£6.50
Advanced BASIC	£6.00
My Computer Likes Me . . . When I Speak in BASIC	£2.75
Introduction to PASCAL	£3.95
SCELBAL — High Level Language + Supplements	£20.00

Accounts Payable and Accounts Receivable	£10.95
Payroll with Cost Accounting	£10.95
General Ledger	£10.95

Basic Software Library:	
Vol 1: Business and Games Programs	£17.50
Vol 2: Maths, Engineering and Statistical Programs	£17.50
Vol 3: Advanced Business Programs	£26.95
Vol 4: General Purpose Programs	£7.95
Vol 5: Experimenters Programs	£7.95
Vol 6: Miniature Business System	£32.50
Vol 7: Chess/Medbil/Wdproc Programs	£26.95

Some Common BASIC Programs	£6.30
Computer Programs that Work (in BASIC)	£2.55
First Book Of KIM	£7.00

8080 Standard Monitor	£9.95
8080 Standard Editor	£9.95
8080 Standard Assembler	£9.95
Special Package: 8080 Assembler, Editor, Monitor	£20.00
Bar Code Loader for 6800, 8080, Z80 and 6502	£1.75
Tiny Assembler for 6800 Systems	£5.75

Magazine Back Issues:	
Micro-6502 Journal	£1.50
Personal Computing	£1.75
Interface Age	£2.25
ROM	£1.75
Dr Dobbs Journal	£1.75
Computer Music Journal	£3.75
People's Computers (recent issues called Recreational Computing	£1.75
BYTE	£2.25
Creative Computing	£1.75
Calculators and Computers	£1.75
Kilobaud — reprints only	TBA
73	£2.25
Magazine Storage Box (Holds 12)	£1.25

THIS LIST CANCELS ALL PREVIOUS PRICE LISTS: EFFECTIVE JANUARY 1979 DUE TO FLUCTUATIONS OF THE DOLLAR, PRICES SUBJECT TO CHANGE WITHOUT NOTICE.

HOW TO ORDER

Please note our prices include postage and packing, but not insurance, if wanted add 12p for every £10 of **books** ordered. Make cheques, PO's etc. payable to:
L.P. Enterprises.
CREDIT CARDS accepted
BARCLAYCARD VISA/ACCESS
DINERS CLUB/AMERICAN EXPRESS
N.B. Diners Club orders cannot be accepted by telephone.
Phone: 01-553 1001 for Credit Card orders (24-hr answering service)

Send to address above — All Orders must be Prepaid
Indicate Payment Method: — Total Enclosed £. . . . .

. . . . My cheque, P.O., I.M.O. is enclosed in **Sterling** on **U.K. Bank**

. . . . Charge to Barclaycard/Visa/Access/Diners/American Express

Credit Card No . . . . Expiry Date . . . .

Name . . . .

Address . . . .

. . . . POSTCODE . . . .

Signature . . . .

All publications are published in U.S.A. and shipped air-freight by **L.P. Enterprises.** In unusual cases, processing may exceed 30 days
Prices subject to change without notice.

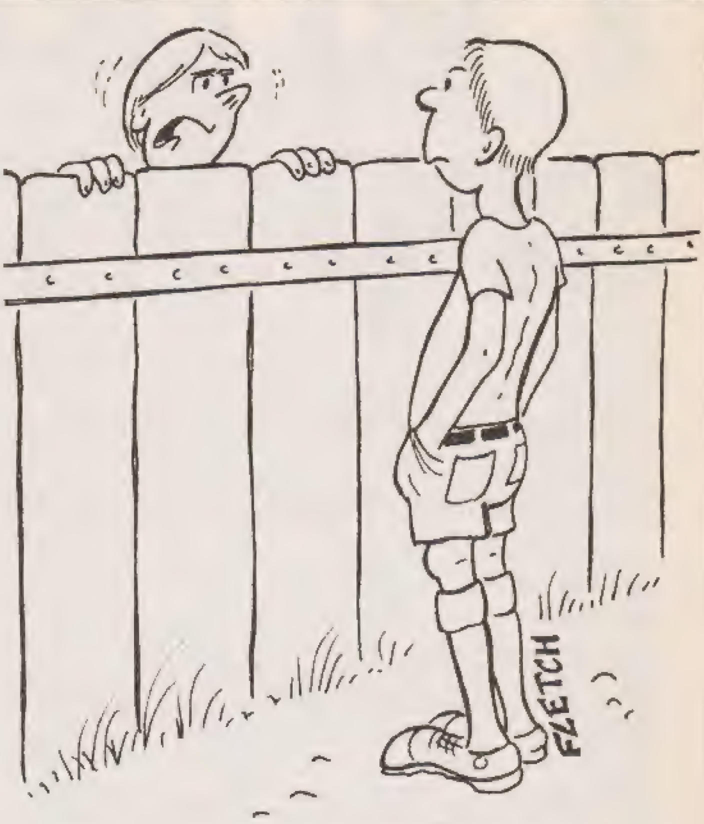
TRADE ENQUIRIES WELCOME

FLETCH

"I've invented silicone chips capable of controlling nuclear fission".

"You mean . . . ."

"Yes — fission chips!"



"An' what's more—my dad's home computer is bigger than your dad's !"

"I'd like to exchange it—my wife doesn't like the colour."

COMPUCOLOR REVIEW

The Computer arrived as two seperate units, the monitor/CPU/disk drive and the keyboard. As we had a demonstration model which was one of the US versions we also had to have a voltage converter but we understand that the UK versions will not need one (check carefully!!). The first impression of the computer is the neatness and quality of the finish, the disk drive being almost invisible.

Comparing this model we received with the one that the author used for a year we found vast improvements in both physical appearance and quality of display. The main ones are that the disk is now built into the monitor casing qnd the screen is much less prone to wobble and mis-alignment.

Hard Facts

The hardware is very well documented in a thick 200 page manual that covers basic operation (with full circuits) interface details and chip descriptions. It is really a service manuall and it would enable a competent engineer to de-bug most faults.

The processor is based on the familiar 8080 and the main circuit board and support circuitry are housed in the monitor case. The board is mounted horizontally in the case and this is possibly one of the really bad points of the system. The board is held in place soley by the back panel and a single slot arrangement.

This is not a very satisfactory arrangement at all, as the board could easily be knocked out of its holders and even rest against the HT — which would not be a pretty sight.

The VDU is a 13" model and has a reasonably high definition. The picture remains quite steady. The alignment was slightly out on our model but not enough to cause any problems. The screen format is 64 characters per line with 32 lines per page.

The ASCII character set is 64 characters with a further 64 specials on a five by seven matrix in a six by eight block. The graphics size may be expanded to 128 by 128 blocks and these facilities are discussed in this review. The cursor can be chosen as either visible or transparent and is non-destructive. Compucolour's CPU is expandable to 64K and has 4K RAM for the screen refresh and 4K RAM for the user workspace. The ROM is 16K on-board and contains the BASIC, the file control system and the terminal software. According to the manual it can be expanded to 28K on-board but we could find no sockets for this so presumably there is a sub assembly that will carry the extra memory.

The outputs from the processor are via edge connectors and this is another untidy point against the machine. The three connectors at the rear of the machine provide the keyboard connection, the modem/ RS232 port and the 50 pin bus output. For neatness and convenience the RS232 should be put on a standard D type socket and possibly the keyboard as well. The built in floppy disk is a 5¼" type with 40 tracks per side. The data transfer rate is 76.8Kb and the disk holds 51.2K per side. The disk must be turned over to access the opposite side's data. The unit is a Wangco type which are well proven and should cause no problems.

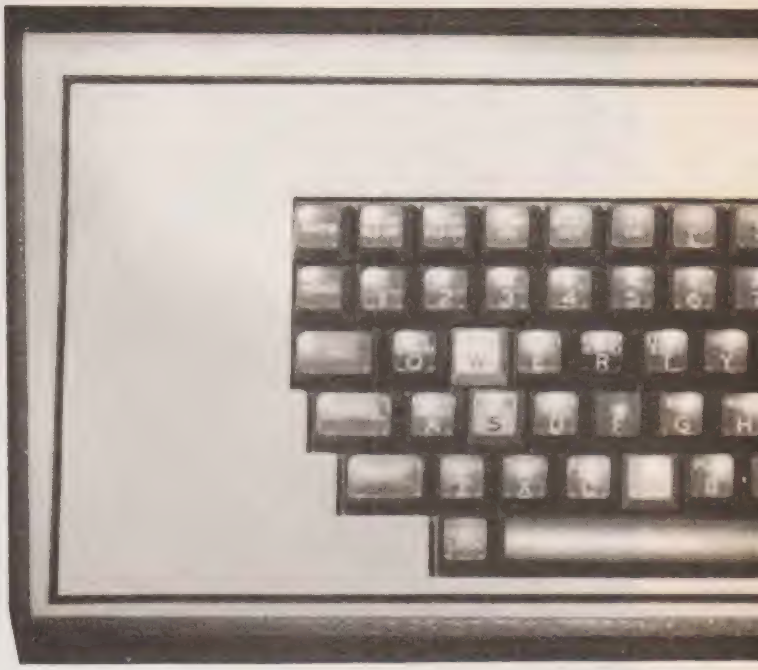
The keyboard is a 72 key version, 101 and 117 key versions are optionally available, and is housed in a sloping front cabinet. The quality of the engraving on the keys is high and the various functions are clearly indicated. Each key can generally do several things by using the ESC and CONTROL functions, and this allows the 192 required codes to be generated. The unit is connected to the CPU by a ribbon cable to the rear edge connector.

An Impressive Start

The author was most fortunate in having actually met one of these machines in an earlier form and was already aware of the power of the operating system. The first indication to of this to a new user is when he tries out the demo disk and it really should be tried before you attempt to use the machine in ernest. The disk is inserted in the drive and the door closed, then all one has to do is press the load key and the disk directory is loaded onto the screen. The American term is MENU and one merely selects an item from this by keying in the number. The program will now load and execute from the start automatically. This capability can be built into any programs that you write and has obvious advantages over a manual system.

The demo disk contains a variety of games and test routines, including a memory test, and these should all be tried to familiarise with the keyboard. The actual programming will be covered later.

The colour and size of the display can also be controlled directly so if you fancy typing in red on a blue background no problem! Having played with the demonstrations the next step is to read the manual on software. It woud be most unwise to miss this out as althouigh it is very easy to program in the BASIC you would be missing out on a very powerful system.



Manual Operations

The Compucolor manual is one of the most thorough sets of software documentation that we've come across in a very long time. It starts where it should with the basics of the BASIC language and progresses to really sophisticated programming techniques. The whole document is well indexed and allows for quick reference to any section. The odd Americanism slips in, MENU for Directory, but at least these are consistent and once understood cause no problems. The only oddity is the fact that they always refer to DISK BASIC and yet it is ROM based. The reason is that the BASIC contains the full Disk Operating System (DOS). The sections on the extensions to the language should be read carefully and the liberal scattering of examples tried out.

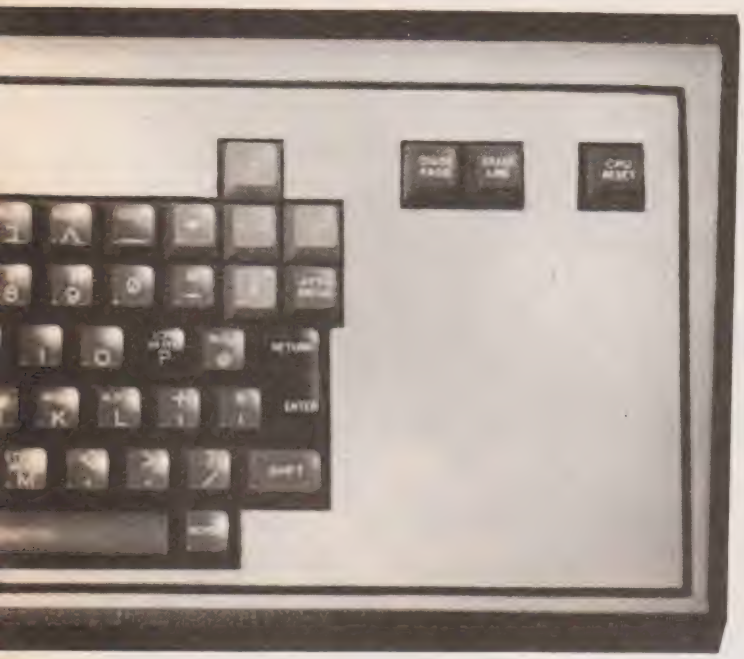
About half of the manual's 150 pages deal with the normal BASIC language. The next section is concerned with the colour, graphics and programmable functions. As a measure of the ease of using the manual it took us about an hour to adapt an example program to try out all the color and graphics facilities continuously in each mode. If that sounds a bit simple to you the program loop to perform all the functions takes about 5 minutes to run through.

The manual has appendices covering all the programmable capabilities and all the error codes, etc. It also gives reasonable details on the interface requirements, machine codes and IC descriptions. This documentation is of very high quality throughout and although not light reading is essential to the user. It is a pity that other machines do not go to these lengths with their manuals.

The BASIC Principles

Once the machine has been turned on and initialised the BASIC may be caled direct from the keyboard. This is done by typing ESC W and the operating system then asks for for memory size. If you require the full memory just type RETURN and size. If you require the full memory just type RETURN and the full on-board allocation is available. This may seem to be a waste of time but it does allow you to dimension your memory to the program size. One excellent feature is the BASIC reset key which can be used in the case of a program fault or runaway and resets the BASIC to where you started without destroying your program. This will even work if you press CPU RESET or come out of DOS so it can save many otherwise embarrassing situations.

The actual BASIC implemented on the system is a full extended version and the command set is listed in Table 1. Unfortunately there is no mention of the origination pof language apart from the fact that it originally came from Dartmouth College. From personal



experience it resembles the Microsoft implementation but suffice to say it is very powerful. It really is almost too powerful for amateur use and would provide for a small business user or even as an intelligent terminal.

The fact that the language contains the full disk operating capability means that data is rapidly stored or retrieved and you don't have to hang around waiting for that program to load as it is about 70 times faster than an average cassette based system.

With the comprehensive manuals and the intelligent way that the operating system has been configured the language is a joy to use and strongly reminds the author of the days when he used to play with BIG machines (PDP8's and 11's).

A Graphic Demonstration

The graphics capability of this machine is very impressive and also quite easy to use. The whole graphics section can be accessed by a single B ASIC command, PLOT, and the available features are laid out in Table 2. Block graphics may be drawn and there are 64 special characters available as well, similar to those on the PET. The relevant section of the software manual really deserves to be expanded and produced as a seperate document as the information is so tightly paC KED IN THAT IT TAKES SEVERAL ATTEMPTS TO READ: It is not lacking in detail, rather that there is too much.

The best approach to tackle each command is to read the section and run the example program that is supplied, for sections with no example try to re-write the previous one to suit. We found out how to change colours and plot modes in about half a day so it's not too difficult really.

The one section which could be clarified a little is the colour change. To do this the PLOT command is followed by a number or numbers which relate to the various PLOT sub-modes that you want to use. This also allows you to output characters direct to the screen, for example:

PLOT 65,66,67,68,69,70.

would print ABCDEF to the screen as the numbers are those which correspond to the ASCII codes.

The colour display allows the background and foreground to be set to any of eight colours. These are Black, Blue, Red, Green, Magenta, Cyan, Yelow and White. The foreground may also be set to blink if required, and the character size set to single or double height as needed. All of these functions may be directly accessed from the keyboard or via BASIC.

The individual colours may be selected by seperate PLOT commands or by using PLOT 6,x where x is the

The BASIS Command Set

Standard Commands:

CLEAR	CONT	DATA
DEF	DIM	END
FOR	GET	GOSUB
IF	INPUT	LIST
LOAD	NEXT	ON
OUT	PLOT	POKE
PRINT	PUT	READ
REM	RESTORE	RETURN
RUN	SAVE	WAIT
TAB	SPC	

FILE"N",FILE"R",FILE"A",FILE"C",
FILE"D",FILE"T",FILE"E"

String Handling Commands:
ASC (XS) , CHRS (X) , LEFTS (XS) , I), LEN (XS) , MIDS (XS , I , J) RIGHTS (XS , I) , STRS (X) and VAL (XS) :

Basic Operatio s:
+,—,=,*,/,↑, NOT,and, or and the following relational tests
> = , = >, « = , = <,><.

Graphics Functions On The Compucolor

Character Plot	This allows you to plot graphics and block characters.
X Point Plot	This plots a point in the X axis.
Y Point Plot	This plots the Y axis point.
XY Incremental Plot	Plots the previous point incremented in a given direction by a given amount.
X Bar Graph, X0 Value	Defines a horizontal bar graph with its starting position and length.
X Bar Graph, Y Value	Plots the Y value of an X Bar Graph.
X Bar Graph, X Max Value	Plots the maximum value of the Bar.
X Incremental Bar Graph	Defines the increment for a plotted Bar Graph.
Y Bar Graph, Y0 Value	As X type but in Y direction.
Y Bar Graph, X Value	As X type.
Y Bar Graph, Y Max Value	As X type.
Y Incremental Bar Graph	As Y type.
X0 Vector Plot	Draws a vector between defined points.
Y0 Vector Plot	As above but in the other plane.
Incremental Vector Plot	Increments the endpoints of the vector by a defined amount.

A7	A6	A5	A4	A3	A2	A1	A0
PLOT	BLINK	BACKGROUND			FOREGROUND		
		BLUE	GREEN	RED	BLUE	GREEN	RED

decimal value of the binary codes selected in Table 3. For example PLOT 6,97 will give you a blinking Red foreground on a Blue background, Yuk! The colours can be mixed from the table, PLOT 5,6 gives youMagenta on a black bachground. The plot mode can also access the special character set directly and place them in any position on the screen. The reference position is the bottom left hand corner and you may travel 127 positions away vertically or horizontally. Thus to put a spot at the screen centre the co-ordinates are 63,63. The four corners, clockwise from bottom left, are 0,0 0,127 127,127 and 127,0.

You can have enormous fun with these graphics as they have been well thought out and are quite easy to understand even in a short time. The only observation that can be made is that for the average hobbyist, if there is such a person, they are probably far too powerful and most of the functions will never be used. For a dedicated games 'freak' this machine will be a graphic success and even a small businessman could use the graphics to good advantage when preparing information.

Summary

Overall the Compucolor is a very fine colour graphics machine that has a supurb extended BASIC and is easily usable. Like all home machines it has its faults but these cannot be found in the software for a change. The main points that really do need to be looked at on the machine are the lack of proper interface connectors and the shoody way the PCBs are mounted in the video monitor.

These two points really do downgrade the appeal of the machine to the person who wants to get in and expand the hadware side of the machine. If you simply want a co?lour graphics computer that is easy to program and is disk based for speed and convenience than this is a serious competitor to the Apple II and slightly cheaper into the bargain. Overall then an impressive machine but badly let down by the construction.

electronics today international

What to look for in the July issue: On sale June 1st

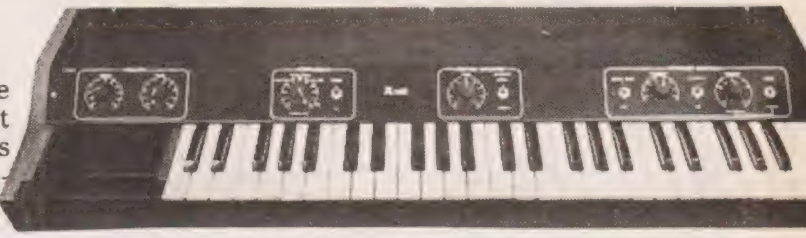
TELETEXT BOX

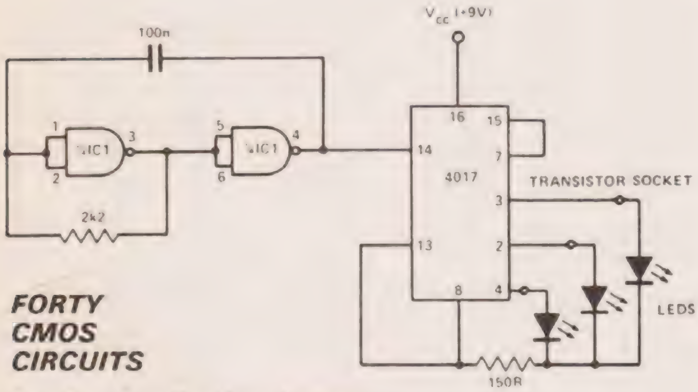
ETI goes Teletext next month. A full spec design including full colour and double height characters. Remote control is by ultrasonics, so there is no need to move from your armchair to change the page. The circuit is based upon the Mullard chip set — long awaited that is.

Emphasis has been placed upon ease of construction — the PCBs are plated through and silk screened and everything mounts to the board. With all this offer and with commercial units running at £200 plus you'd expect this kit to cost the earth would you not? Well it will set you back under £160 complete and we don't think that's bad! Don't miss this.

POLYPHONIC KEYBOARD CONTROLLER

We've struck the right chord here. Give up those one note wonders and take up polyphonics — you can't get arrested for it and it'll make your oscillators warble for joy. Play away up to 8 times simultaneously and don't feel guilty about it!





FORTY CMOS CIRCUITS

Another family size, bumper bundle of goodies from our bionic project editor, Ray Marston. In past issues he's covered 555s and 741s. This time it's the turn of CMOS circuits. He's got forty, yes forty, of the little beauties for you. You can't afford to miss them.

SOIL MOISTURE INDICATOR

Don't drown your Dahlias. The most common cause of premature plant death (murder) in the home is over-watering. If you don't want the Sweeney hammering on your door in the middle of the night, we strongly suggest that you build this moisture indicator. Think of it as a happiness guide for your busy Lizzie.

Mr. J.F. Taylor

Nim's Game

This program is a calculator version of Nim's game. The object of the game is to force your opponent to take the last match. You may initially specify the number of matches in the pile, the maximum number of matches that can be taken from the pile in any one go. There is also a choice of going first or second.

After some matches have been taken from the pile, you are told the number left. The calculator then makes its move. Zero and numbers greater than the maximum specified are illegal moves and the player will be requested to re-input the number of matches to be taken away from the pile. Negative numbers are treated as positive ones and fractions are truncated.

The program, in its present form, can only be run on a Texan TI59 calculator with printer because of the large number of memories used. If the printer were not used, the number of registers used could be reduced to 5 (00 – 04), but then parts of the program dealing with the printer would have to be omitted. This should cause no problem as there is no absolute addressing in the program. Once the program and date constant have been entered, it is advisable to put them onto magnetic cards (2 required) to save having to type the date again.

To start the program press 'A' for a header on the printer, otherwise 'B'. Then after each number is entered, press 'R/S', and the number is accepted and printed on the printer. In response to the question 'FIRST', a reply of 0 (nought) instruct the calculator to make the first move, any other number enables the player to go first.

PROGRAM LISTING

```
000  76 LBL
001  11  A
002  05  5
003  71 SBR
004  97 DSZ
005  09  9
006  71 SBR
007  97 DSZ
008  01  1
009  03  3
010  71 SBR
011  97 DSZ
012  98 ADV
013  98 ADV
014  98 ADV
015  76 LBL
016  12  B
017  01  1
018  07  7
019  71 SBR
020  97 DSZ
021  91 R/S
022  59 INT
023  50 IxI
024  42 STO
025  00  00
026  99 PRT
027  76 LBL
028  42 STO
029  02  2
030  01  1
031  71 SBR
032  97 DSZ
033  43 RCL
034  00  00
035  32 X:T
036  91 R/S
037  59 INT
038  50 IxI
039  42 STO
040  01  01
041  99 PRT
042  77  GE
043  42 STO
044  02  2
045  05  5
046  71 SBR
047  97 DSZ
048  91 R/S
049  59 INT
050  50 IxI
051  99 PRT
052  29 CP
053  98 ADV
054  98 ADV
055  67  EQ
056  23 LNX
057  76 LBL
058  28 LOG
059  02  2
060  09  9
061  71 SBR
062  97 DSZ
063  43 RCL
064  01  01
065  85  +
066  01  1
067  95  =
068  32 X:T
069  25 CLR
070  91 R/S
071  59 INT
072  50 IxI
073  42 STO
074  02  02
075  99 PRT
076  77  GE
077  28 LOG
078  29 CP
079  67  EQ
080  28 LOG
081  22 INV
082  44 SUM
083  00  00
084  03  3
085  03  3
086  71 SBR
087  97 DSZ
088  02  2
089  32 X:T
090  43 RCL
091  00  00
092  99 PRT
093  22 INV
094  77  GE
095  24 CE
096  76 LBL
097  23 LNX
098  01  1
099  42 STO
100  54  54
101  76 LBL
102  35 1/X
103  43 RCL
104  01  01
105  32 X:T
106  43 RCL
107  00  00
108  75  -
109  43 RCL
110  54  54
111  75  -
112  01  1
113  95  =
114  22 INV
115  77  GE
116  25 CLR
117  43 RCL
118  01  01
119  85  +
120  01  1
121  95  =
122  44 SUM
123  54  54
124  43 RCL
125  54  54
126  32 X:T
127  43 RCL
128  00  00
129  77  GE
130  35 1/X
131  76 LBL
132  25 CLR
133  43 RCL
134  54  54
135  32 X:T
136  43 RCL
137  00  00
138  67  EQ
139  34 ΓX
140  75  -
141  43 RCL
142  54  54
143  95  =
144  42 STO
145  53  53
146  29 CP
147  43 RCL
148  53  53
149  22 INV
150  67  EQ
151  33 X²
152  76 LBL
153  34 ΓX
154  02  2
155  42 STO
```

```
156   53   53      200   42  STO
157   76  LBL      201   03   03
158   33  X²       202   05    5
159   03    3      203   32  X:T
160   07    7      204   01    1
161   71  SBR      205   42  STO
162   97  DSZ      206   04   04
163   43  RCL      207   69  OP
164   53   53      208   00   00
165   99  PRT      209   76  LBL
166   22  INV      210   54   )
167   44  SUM      211   73  RC*
168   00   00      212   03   03
169   04    4      213   84  OP*
170   01    1      214   04   04
171   71  SBR      215   69  OP
172   97  DSZ      216   23   23
173   02    2      217   69  OP
174   32  X:T      218   24   24
175   43  RCL      219   43  RCL
176   00   00      220   04   04
177   99  PRT      221   22  INV
178   77   GE      222   67   EQ
179   28  LOG      223   54   )
180   98  ADV      224   69  OP
181   98  ADV      225   05   05
182   04    4      226   92  RTN
183   05    5      227   00    0
184   71  SBR      228   00    0
185   97  DSZ
186   61  GTO
187   12    B
188   76  LBL
189   24  CE
190   98  ADV
191   98  ADV
192   04    4
193   09    9
194   71  SBR
195   97  DSZ
196   61  GTO
197   12    B
198   76  LBL
199   97  DSZ
```

LABELS USED

001	11	A
016	12	B
028	42	STO
058	28	LOG
097	23	LNX
102	35	1/X
132	25	CLR
153	34	ΓX
158	33	X²
189	24	CE
199	97	DSZ
210	54	)

CONTENTS OF REGISTERS

1.	00
4.	01
4.	02
21.	03
5.	04
3723243600.	05
3335322235.	06
1330003327.	07
1345360000.	08
3124303036.	09
2213301700.	10
0.	11
0.	12
2020202020.	13
2020202020.	14
2020202020.	15
2020202020.	16
3637133537.	17
3132400000.	18
0.	19
0.	20
3013440031.	21
3200000000.	22
0.	23
0.	24
2124353637.	25
0.	26
0.	27
0.	28
3713261700.	29
3221210000.	30
0.	31
0.	32
3132400027.	33
1721370064.	34
0.	35
0.	36
2400371326.	37
1700000000.	38
0.	39
0.	40
2717134224.	41
3122000000.	42
0.	43
0.	44
4532410027.	45
3232361700.	46
0.	47
0.	48
4532410043.	49
2431000000.	50
0.	51
0.	52
1.	53
1.	54
0.	55
0.	56
0.	57
0.	58
0.	59

SAMPLE GAMES

```
THIS PROGRAM
PLAYS
NIMMSGAME
STARTNO.
              15.
MAX NO
               3.
FIRST
               1.
TAKE OFF
               3.
NO. LEFT =
              12.
I TAKE
               3.
LEAVING
               9.
TAKE OFF
               2.
NO. LEFT =
               7.
I TAKE
               2.
LEAVING
               5.
TAKE OFF
               0.
TAKE OFF
               5.
TAKE OFF
               3.
NO. LEFT =
               2.
I TAKE
               1.
LEAVING
               1.
YOU LOOSE
STARTNO.
               5.
MAX NO
               4.
FIRST
               1.
TAKE OFF
               4.
NO. LEFT =
               1.
YOU WIN
STARTNO.
```

Mr. B. Owen

HUMBUG MONITOR

A new, more powerful monitor for TRITON with many added features

The original TRITON monitor program was written to give the machine code programmer a tool to enter, modify and run programs. It provided routines to drive the keyboard and VDU for Tiny BASIC and routines to enable the user to dump and load programs from a bulk storage medium, namely cassette tape. This first monitor was written to fit into a 1K EPROM (2708) and so it could only provide limited facilities. The author decided soon after building his TRITON that a better monitor was needed to develop machine code programs. At the same time Don Scales (who configured Tiny BASIC for TRITON) decided to use part of the fourth EPROM to extend the facilities of Tiny BASIC. It was therefore a logical step to write an extended monitor for the TRITON using the rest of this EPROM. ROMBUS 5.1 is the firmware package incorporating the upgraded BASIC and HUMBUG. While HUMBUG does not provide all the facilities one would like in a monitor program (eg single step execution), it goes a long way to making life easier for the programmer. The extended Tiny BASIC allows machine code subroutines to be called, so to make the most of your TRITON you will probably want to develop machine code subroutines with the aid of the monitor.

New Functions

The new monitor contains the original seven functions and eight new ones in addition. The new functions are register dump and modify after a breakpoint, continue from a breakpoint, ASCII string creation and display, formatted hexadecimal dump, tape motor on/off control, base conversion between decimal and hexadecimal and vice versa. The final function allows a choice of output device between the VDU and a serial output port. The device onto which output appears depends on the contents of memory location 1401H. In addition hitting the Reset button does not cause the memory automatically to be cleared as we shall see.

The new monitor resides in Read Only Memory (EPROM) from locations 0000H up to 03FFH and from 0DBEH up to 0FFFH. When you switch your TRITON on a power up reset is generated. This is a signal to the computer to start obeying instructions from location 0000H. At this location the computer finds the instructions which set it up ready for you to use. The first of these initialises the stack pointer. This is a pointer to an area of memory which is used to store parameters and return addresses for subroutines. The program then selects the VDU as the output device, clears the screen and announces itself saying:

INITIALISE?

The monitor now waits for a response from you. If the response is 'Y', your TRITON executes a memory test routine. This does a checker board memory test. This involves storing the bit patterns 01010101 and 10101010 in every memory location and checking that each pattern is correctly stored. When a read back error occurs, the address is stored in locations 1481 and 1482H. This address is used by BASIC to determine the size of the workspace available. The routine then initialises the variable (1402,3H) which determines the Baud rate of the serial interface to 110 Baud and sets up a jump table which is used by Tiny BASIC to access the input and output routines in the monitor.

On returning from this routine or if any response other than 'Y' is typed, the prompt:

FUNCTION? P G I O L W T R C A D H V M B

IS PRINTED: When you type in a character the TRITON checks it against those in the prompting list, and if a match occurs, it jumps to a routine to obey the function. Otherwise the message:

INVALID

is printed and the TRITON waits for another character which it processes in the same way. When a function terminates the TRITON re-displays the prompting message listing the functions available. In some cases if an invalid character is received by one of the functions, the INVALID message will appear and the TRITON will wait for a new function to be entered.

Subroutines AndFacilities

Most of the subroutines from monitor V4.1 have been retained but they have all been relocated in order to obtain a more compact program. All of the standard monitor utilities have been maintained and a new one has been created. This new one, called ECHOCH is used by means of a CALL instruction and it fetches a character from the keyboard, echoes it on the selected output device and returns the ASCII code for the character in the accumulator.

The characteristics of the input and output subroutines have been changed slightly. As you will know if you have used version 4.1 monitor, the TRITON accepts unshifted letters as upper case letters and shifted letters as graphic characters. In the old input routine blocks of 32 characters were shifted by the software. This meant that the symbols , [,] , ând required unexpected action of the shift key in order to access them and their corresponding graphics. With HUMBUG this action is not needed and all characters are accessed as depicted on the keyboard. (ie requires shift to be depressed while [and] do not.)

The character output routine checks the keyboard to see if CONTROL S has been typed before it outputs a character. In the ASCII code this character is called X-OFF and it is used here to temporarily suspend output. If CONTROL S has not been typed, output continues as normal, but if it has the character will not be displayed. Instead the TRITON will do nothing except wait for you to type another character. If the character is CONTROL C, then the computer will re-initialise with the function prompt. If you type CONTROL Q (ASCII code for X-ON) the output will resume with the current character and then continue as normal until the output ends or another CONTROL S is received. If you type any other character it is ignored. This facility to interrupt the output stream is particularly useful when you are using the hex dump function or

listing a BASIC program in order to give you time to read the information before it scrolls off the screen.

You can abort from most of the functions by typing CONTROL C although the computer must be expectring an input character when you type this. This is not the case during the tape input and output functions after a tape has been started, if a machine code program gets stuck in a loop and during a hex dump. You can abort the latter by typing CONTROL S (to stop the output) and then CONTROL C to quit. To quit from the first three cases you can use interrupt 2 to reset. It is alright to use reset since HUMBUG will not clear memory unless you instruct it to, as explained above. If you have used interrupt 2 you will have noticed that your TRITON prints some more information besides FUNCTION etc. This information is a list of the contents of all the 8080 registers after it obeyed the instruction during which you hit the interrupt button. This is especially useful when you use the interrupt to get your TRITON out of a loop because the contents of the program counter tell you where the loops in your program. The program counter is a pointer used by the 8080 to tell it the address in memory of the next instruction to be obeyed. Every time an instruction is executed the contents of this pointer are updated. A typical display after using interrupt 2 is:

```
F A C B E D L H SP PC
46 20 00 00 14 14 00 38  1470 39A7
FUNCTION? P G I O L W T R C A D H V M B
```

Other Major Differences

There are several other important differences to note in HUMBUG compared with monitor V4.1. When you switch your TRITON on or execute a reset and ask HUMBUG to initialise, the memory will not end up filled with 00H in every byte as before. Instead even addressed bytes (eg 1600H) will contain AAH and the odd addressed bytes will contain 55H. This is because the new memory test subroutine uses a checker board technique as explained above to test for short circuits between adjacent data bits. It does not matter that these patterns are left in memory and they will not affect your programs when you enter them. The stack pointer of the monitor has been changed from 1480H to 1470H so that the sixteen bytes between these values can be used by the BASIC to permit greater flexibility. If you have used 1480H as the initial value of the stack pointer in any of your programs, you should change it to 1470H. Finally the result of interrupts 3-7 is a jump to a location between 1430H and 143FH instead of a jump to somewhere between 1618H and 1638H. This has been done to allow you to use interrupts in BASIC programs without having the problem of the interrupts being vectored into your BASIC source statements. If you have already written programs utilising these interrupts or the RST 3-7 instructions you must set up the vectors at the addresses in the table below before running your program.

It was mentioned earlier that a serial output port has been provided. This utilises bit 8 of output 06H as a serial output line. The data is formatted into a serial fashion using software routine. The description given above for the character output routine was slightly simplified. After checking the keyboard for CONTROL S, the routine checks the contents of location 1401H. This location is called the display switch and if it contains 55H then the output is directed to the serial output port. If there is any other value then output occurs on the VDU. Having assertained where the output must be directed the routine performs the appropriate operations. The serial output port produces a pulse train using negative logic (ie a Zero is represented as +5V and a one as 0V). The framing format is one start bit (zero) and two stop bits (one's) as shown in Figure 1.

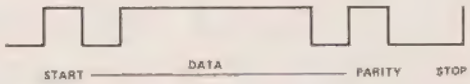


Figure 1 Output format for ASCII 'A'

A fake parity bit is generated for the last bit of the eight bit data field. This format is suitable for TELETYPES etc. where parity checking is not required. To allow time for the mechanics of a printer to perform a carriage return cycle a delay of three character periods is introduced after a carriage return code is sent to the serial port. During the delay a continuous stop level is sent. When the computer is initiated the bit time is set to 9.09 mS which corresponds to a bit rate of 110 Baud or a data rate of 10 characters/second. If you wish to use a printer or VDU at any other speed it is a simple matter to change the speed by changing the sixteen bit value in locations 1402-3H. The required value can be calculated from the following formulae.

$$T_{bit} = \frac{1}{\text{Baud rate}} \text{ sec}$$

$$t_{cy} = \frac{9}{\text{clock rate}} \text{ sec}$$

$$t_{cy} = 1.255\text{uS for 7.16MHz clock}$$

$$n = \frac{\frac{T_{bit}}{t_{cy}} - 107}{24}$$

eg for 110 Baud

$$T_{bit} = \frac{1}{110} = 9.09\text{mS}$$

$$n = \frac{\frac{9.09\text{mS}}{1.26\text{uS}} - 107}{24}$$

$$= 297 \quad \text{or } 0129\text{H}$$

The minimum value of n is 001H which corresponds to about 6K Baud. Hopefully by now you will have realised that it would be possible to change from outputing on the VDU to outputing on a printer whilst your programs are running. Listed below in ,istings 1 and 2 are the machine code and BASIC instructions to do this. The machine code version includes the hex instruction codes.

```
3E  55       MVI  A,55H
32  01  14   STA  A,1401H ;SET PRINTER MODE

3E  AA       MVI  A,AAH
32  01  14   STA  A,1401H ;SET VDU MODE
```

Listing 1 Machine code instructions to change o/p mode

```
a = 21760
POKE 5120,A          ;REM SET PRINTER MODE
```

```
B = 22016
POKE 5120,B          ;REM SET VDU MODE
```

Listing 2 BASIC instructions to change o/p mode

Obviously the BASIC instructions can use any variables but you are not allowed to poke a constant. When the POKE 5120 instructions are used location 1400H has 00H written into it. This is the same as the value put there when the character input routine is entered. If you wish to use the serial output facility you must connect an interface between the output port and your peripheral device. It is quite likely that you will require either an RS232C or 20 mA loop signal and circuits to generate these are given below in Figure 2. For other interface signals you will have to design your interface. In this case remember that the port produces negative logic.

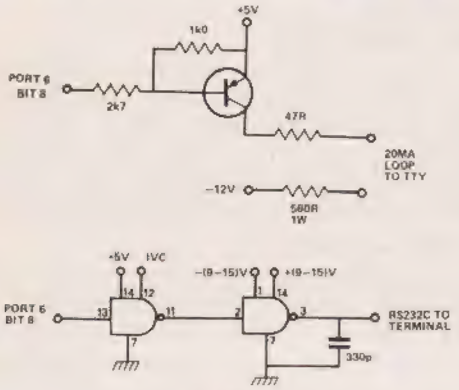


Figure 2

We will now look at the functions in turn and see exactly how to use each one. Except for the R and B functions you only need to type a single letter to invoke a function. You should not type carriage return.

P — Program
The purpose of this function is to allow you to enter machine code programs into your TRITON's memory. When you type P in response to the function prompt, HUMBUG will respond with:

PROG START = .

You should now reply with four hex digits to specify the address at which you wish to start programming, followed by carriage return. If you type a wrong character you can correct it by typing CONTROL H until the cursor points to the wrong character. You must then retype the rest of the address again. When you type carriage return, HUMBUG will print the address on a new line followed by a space and two hex digits to represent the data held in that address and finally another space. Your TRITON will now wait for you. You can modify the contents of the memory location by typing two hex digits followed by carriage return. Once again errors can be corrected by typing CONTROL H and retyping the character(s) deleted. Whether you change the contents of the memory location or not, you have the choice of stepping on to the next location by typing carriage return or stepping back to the previous location by typing up arrow (SHIFT^). This facility gives you the chance to correct errors in a memory location after you have moved on to another location. When you do not want to change the contents simply type carriage return or up arrow depending on which location you want to open next. After you have finished type CONTROL C to quit back to the monitor loop.

G — Go To Program
This function enables you to start programs anywhere in memory. After typing G, the prompt:

PROG START =

is displayed. Once again you must specify a four digit hex address and this can be corrected as before. After you have typed carriage return, your TRITON will immediately jump to the address specified. (eg Type G 0400 and your TRITON will jump to Tiny BASIC which starts at address 0400H.)

I — Input from cassette tape
The function enables you to input a program from a previously recorded audio tape . After typing I, HUMBUG will ask you for the header code of the program that you wish to load with the prompt.

TAPE HEADER = :

When you have typed the name and carriage return, the tape recorder motor is started automatically and then FILES FOUND: is printed on display. When a header code is found it is printed on the next line. If this is a perfect match with the name that you specified, then the cursor will stay on the same line and the program will be loaded to 1600H. At the end of the file, the motor is turned off and HUMBUG will say END then re-initialise with the FUNCTION? prompt. If the header code is different the cursor will move to the beginning of the next line and the computer will search for the next header code which does not exist on the tape (eg just header code which does not exist omn the tape (eg just type carriage return) you can produce a list of the programs stored on the tape. The programs are recorded onto the tape at 300 Baud.

O — Output a program to tape
This is complementry to the above function and it allows you to dump programs on tape. Once again you have to specify the tape header code. This code is recorded onto tape as an identifier before the program. BASIC programs are recorded onto tape automatically but you must set the programs and address for machine code programs before you enter the O function. This involves putting the address of the first location after your program into 1600 and 1601H in the normal 8080 format with the least significant byte first. When the whole program has been dumped, HUMBUG responds with END and the function list prompt. The recording format is:

About 5-6 seconds of mark tone
64 bytes of leader code (0DH)
Program end address
Program
About 5-6 seconds of mark tone

L — List memory
This function allows you to list the contents of fourteen memory locations at once, but not change them. HUMBUG asks you for a start address and after you have

typed an address and carriage return your TRITON will print the address and contents of the next fourteen locatiions down the left hand side of the display. When the listing stops, you will see the prompt MORE? at the end. If you type Y(yes), the contents of the next fourteen locations are printed and so on. If instead you type one of the function letters, that function will immediately be invoked. Typing anything else will cause HUMBUG to display the INVALID prompt.

W — Typewriter mode

After typing W your TRITON will act as if it were only a keyboard connected to a display. Everything you type is echoed onto the selected output device. When you use the VDU all the graphics are displayed and the cursor control codes act as normal. To get out of this mode you must type CONTROL C.

T — Tiny BASIC

When you type T your TRITON executes a jump to Tiny BASIC and you will see the message:

BASIC L5.1
OK

You should read the seperate section to find out how to use the BASIC interpreter.

R — Register dump and modify

This function is probably the most useful one for debugging machine code programs. After typing R nothing will appear to happen. In fact the computer is waiting for you to type another character. If you now type SPACE the register contents will be dumped in the same format as described earlier in connection with interrupt 2. It should be realised that these are not the contents of the 8080 registers at the present moment but the contents of a set of virtual registers. They were the contents of the registers last time an interrupt 2 occured.

To modify any register you must type the letter that represents the register as the second letter of the function call. For the processor status word (flags) use F, for the stack pointer use S, and for the program counter use P. The display will show a two or four digit hex code as appropriate followed by space. If you wish to change the value you can type in a new hex number of the correct number of digits (the same number as displayed). Alternatively, to leave the contents unchanged, simply type carriage return. You would normally use this function in conjunction with the continue function and breakpoints. We shall see how later.

C — Continue

This function loads the values of the virtual registers into the 8080 machine registers. Since the program counter is one of the registers which is loaded, the computer effectively jumps to the value put into the virtual program counter. If you wish to run a piece of program a number of times you can set up the value of RP as the start address of the program amd simply press C each time to run the program. A more important use of this facility is when you want to test a part of a program starting with values which would have been left in the registers by the previous part of the program. To do this use Rx commands to set up the registers and the proigram counter and type C to start.

A — ASCII string insert

The use of this function is to put ASCII strings into memory for use by machine code programs. Typically you will want to have prompts, etc. displayed by your programs and this allows you to insert the ASCII strings without having to look up the hex code for each character. After you type A, HUMBUG will ask for the starting address of the string. When you have entered this the cursor will step onto the next line. Anything you now type, including cursor control codes, will be stored in memory and echoed back. CONTROL H will delete characters from memory. When you have finished type CONTROL D (EOT code). The EOT code will be stored in memory and then your TRITON will re-initialise. If you do not want the EOT code to be stored, terminate the string with CONTROL C which will not be stored either. The EOT code allows you to display the strings by calling PDATA from your programs.

D — Display ASCII string

This is the reverse of the A function. It allows you to print out the ASCII string starting at the address that you specify. This enables you to check that you have entered strings correctly. You should not use this function for strings that do not end with EOT code, because this function does not terminate until it finds this code.

H — Hexadecimal dump

This function produces a formatted hex dump of a section of memory. After typing H, the computer asks for the start address in the normal way. When you have entered this a request is made for an end address with the prompt: PROG END = .

After you have specified this, the start address is printed at the start of the next line, followed by up to sixteen data bytes. The number of data bytes on the first line is such that the next line starts at address xyz0H. On each line of the dump the address of the first byte on the line is printed followed by the sixteen data bytes. The function is a useful way of obtaining a printed listing of a machine code program.

V — VDU switch

This function is used to select the serial output port when output is on the VDU and vice-versa. After you have typed V nothing else happens on the current output device until it is reselected. The FUNCTION? prompt is printed on the newly selected output device. You can then carry on as normal with all output appearing on the newly selected output device but note the restriction above for the tape input routine. If the function fails to work it means that the data in the display switch is corrupted. Set the value in location 1401H to AAH and try again.

M — Motor control

In the original TRITON design the tape recorder was automatically controlled by the tape routines and there was a manual overide facility for fast winding. The automatic control of the tape recorder has been retained in HUMBUG but the over-ride facility has been incorporated into the software. If the motor is turned off and you type M, the motor will be turned on and vice-versa. This facility is particularly useful when you try to input from tape a non-existent file and have to abort by using a reset or interrupt 2. Doing this leaves the tape motor running. Typing M will stop the motor. This facility makes the front panel over-ride switch redundant so you can disconnect the

switch and use it for one of two things. Firstly it can be used as a pause button and secondly it can be used for interrupt 3.

B — Base Conversion

This function is provided mainly for the convenience of the BASIC language programmer. The PEEK, POKE' READ, WRITE and CALL commands each require an address which must be specified in decimal. It is most likely that the address of data that you know will be in hex so this routine, which works with sixteen bit two's complement numbers, saves you the trouble of carrying out the calculation on paper in binary! After typing B the computer will wait for you to type D or H. This second letter represents the base to which you wish to convert. Hence to convert from hex to decimal type D. Any other letter that you type will be flagged as an error. If you typed a D, HUMBUG will be expecting you to type a four digit hex number. If you wish to convert a two digit number, you must insert two leading zeroes. When you type carriage return the decimal equivalent of the number displayed including a negative sign where appropriate. If you typed H, HUMBUG will expect a decimal number to be input in free form. This means that you can specify a sign if necessary and need not specify leading zeroes. Try BH −1 and you should get FFFF as the hex value returned. The formatting of numbers has been made compatible with the representation of sixteen bit numbers elsewhere in ROMBUS 5.1 (ie four digit hex numbers in HUMBUG and free format decimal numbers in BASIC).

Breaking And Entering

We have now seen what all of HUMBUG's functions can do and we will now see how to make use of them to debug a program. Enter the program shown in listing 3 eactly as shown, (yes it has got an error in it!), using the P function. Try using the up arrow facility (SHIFT ^) to step back down memory at some point. After you have entered it obtain a hex dump using the H function. The start address should be 1600H and the end address should be 1610H. Note that you can stop the output by typing CONTROL S and restart it by typing CONTROL Q. Now carefully check the listing given against the dump you have obtained. If there are any errors go back and correct them with the P function. The aim of the program is to clear the VDU screen and then print the alphabet on the top line of the screen. Now using the A function, enter the alphabet from A-Z starting at 1700H. Terminate the string by typing CONTROL D. Type G 1600 and you should see the screen clear and the letter A appear in the top left hand corner and the re-initialisation message appear on the next line. This is not what the program should do and we must now debug it to get it working correctly. Clearly the jump to NXTCHR is not being executed. We will now set a breakpoint so that we can see what is happening when this point in the program is reached. A breakpoint is a way of stopping execution of a program at a particular point and returning to the monitor in such a way that the previous execution of the program can be ascertained. As we have already seen an interrupt 2 causes the states of all the registers to be saved and displayed on the scren. The interrupt signal can be generated by software using the RST 2 instruction (D7H). Because interrupt 2 causes the registers to be dumped into the virtual registers you must not use this interrupt to terminate a function when you are working with breakpoints or you will lose the values of the registers used by the program. Use CONTROL C instead. It is a good idea to get into a habit of doing this always. To set the breakpoint use the P function to change the contents of address 160BH from CAH to D7H. Now start the program again from 1600H. Again the screen will clear and the A will be printed and then the registers will be dumped as described before. The registers that are of interest are A.F.D. and E. You will see that the stack pointer is set to 1470H (the monitor stack area) by default. The program counter is pointing to the address of the RST 2 instruction plus one (160CH). The value in the DE register pair is 1701H as we would expect. (We loaded it with 1700H and incremented it once.) The value of the accumulator is 41H which is the value loaded fro the data buffer. Obviously 41H is not equal to 5AH and if we look at the flag byte we will see that the zero flag is cleared to indicate this. The format of the flag byte is:

SIGN ZERO xx AUX CY xx PARITY xx Carry

Clearly the sequence of execution will not be affected by the jump if zero instruction. We must change this instruction to a jump if not zero. Using the P function again change the instruction at 160BH to C2H. Now using the RP function change the program counter to 160BH the address we wish to continue from. By typing R SPACE yoiu can confirm that A = 41. PC = 160B, D = 17 and E = 01H. If we now type C the register values will be reloaded and the program will continue from 160B as if it had never been interrupted. The computer encounters the JNZ instruction and since the flags indicate non-zero the jump will be executed. What you should see is the C that you just typed followed by the alphabet from B-Z and then the re-initialisation message on the next line. If this is what you see then the program works (It has already done the rest before we interrupted it.) and if you type G 1600 the program will do what we said it would.

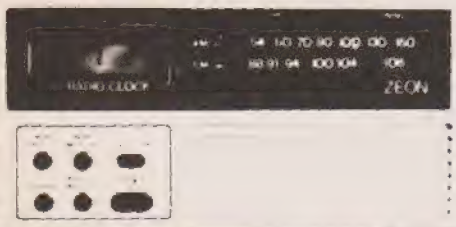
```
1600          RST 1          CF   ; CLEAR SCREEN
1601          LXI D.1700H    11   ; SET POINTER
1602          —              00   ; TO DATA
1603          —              17
1604 NXTCHR:  LDAX D         1A   ; GET CHAR
1605          INX D          13   ; BUMP PTR
1606          CALL OUTCH     CD   ; PRINT A
1607          —              13   ; CHARACTER
1608          —              00
1609          CPI 5AH        FE   ; WAS IT Z?
160A          -              5A
160B          JZ NXTCHR      CA   ; NO SO GET
160C          —              04   ; NEXT LETTER
160D          —              16
160E          JMP START      C3   ; YES SO GO
160F          —              60   ; MONITOR
1610          —              00
```

Implementation of ROMBUS 5.1

ROMBUS 5.1 consists of a set of four EPROMSwhich are available ready programmed from TRANSAM COMPONENTS LTD. at 12, Chapel Street, London, N.W.1. To obtain these you should send your old EPROMS back and they will be reprogrammed with the new software and a fourth EPROM supplied. The EPROMS are labeled HUMBUG A and B and BASIC A and B. The BASIC should be plugged into the socket where MONITOR V4.1 was and HUMBUG B should be plugged into the fourth EPROM socket.

CT MARKETPLACE

CLOCK RADIO



You probably won't believe us as we're selling the goods but we're going to tell you anyway! We have *rejected* eight clock radios for Marketplace, they were all cheap enough but the quality was so poor that we couldn't have lent our name to them. However, we are now able to offer a portable LCD Clock Radio to you which meets our standards.

The clock is a 12-hour one with AM/PM indicated and a back light. The radio is Medium Wave and FM with very nice quality for a small speaker — for FM there's a telescopic aerial. The alarm can be either a 'beep-beep' type or the radio, there's also a snooze facility.

The case is sensibly rugged and is printed on the back with a World Time Zones map, a bit of a cheek really, especially as the time is relative to Japan!

We won't even mention the RRP — but just check on comparable prices — you'll find ours a bargain.

An example of this Clock Radio can be seen and examined at our Oxford Street offices.

£20.50

(Inclusive of VAT and Postage),

To:
CLOCK RADIO Offer,
ETI Magazine,
25-27 Oxford Street, London W1R 1RF.

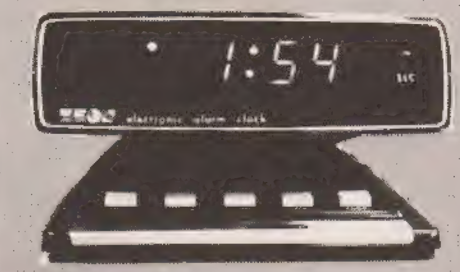
Please find enclosed my cheque/PO for £20.50 (payable to CT Magazine) for my Clock Radio.

Name ....

Address ....

Please allow 28 days for delivery.

DIGITAL ALARM MK2



Both ETI and Hobby Electronics have sold a lot of digital alarm clocks — over 10,000 in fact — maybe that's something to do with the fact that we sell at real bargain prices. Now we can offer you a truly modern, space age model.

It includes all the facilities expected in a good design — fast, slow setting, snooze facility, etc plus two unusual features — automatic brightness control and a weekend alarm cancel.

An example of this clock can be seen and examined at our Oxford Street offices.

£10.50

(Inclusive of VAT and Postage)

To:
DIGITAL ALARM CLOCK MK2 Offer,
CT Magazine,
25-27 Oxford Street,
London W1R 1RF.

Please find enclosed my cheque/PO for £10.50 (payable to CT Magazine) for my digital alarm clock.

Name ....

Address ....

Please allow 28 days for delivery.

ALARM-CHRONO LCD

Currently this watch is being discounted elsewhere for typically £39.95 (we don't quote RRP as this is meaningless) and the watch is a 'Chinese copy' of a very famous one in the £100 range!

The facilities are exceptional:

- Normal hours and minutes
- Continuous seconds or data display
- Day of the week
- Stopwatch with 0.1 secs resolution
- Lap time facility with automatic return to stopwatch after 6 seconds
- Different time zone setting with independent date, day of week settings
- Good bleeping alarm
- Easy time correcting: on the sixth 'pip', press a button and it's reset to 00 seconds as long as watch is plus or minus 29 seconds.

It comes with a full guarantee of course.

An example of this watch can be seen and examined at our Oxford Street offices.

£27.95

(Inclusive of VAT and Postage)

To:
ALARM/CHRONO LCD WATCH Offer,
CT Magazine,
25-27 Oxford Street,
London W1R 1RF.

Please find enclosed my cheque/PO for £27.95 (payable to CT Magazine) for my Alarm/Chrono LCD watch.

Name ....

Address ....

Please allow up to 28 days for delivery.

Another adaptor for the PET that allows you to connect to a serial device

A couple of months ago we received the Petsoft CMC1200 IEEE to RS232 adaptor. Well this month we received another from a London company called 3D. As we had by now built our Heathkit printer it seemed an ideal opportunity to tesdt both of them out. Although this review is primarily about the adaptor we have made one or two comments on the printer as well. We hope to publish a full report on the Heathkit at a later date.

What Your Money Buys

The interface adaptor comes in a single black box about 8" by 4½" by 2½" and is complete with all the necessary leads and plugs, etc. Inside the unit is a mains transformer and all the interface circuitry necessary to givbe either an RS1200 and these worked quite happily with the printer. (The Heathkit can actually go up to 9600 Baud but the manual advises you not to use it)..

The PCB is of commercial quality and neatly fits into the case with the various adjustments nicely to hand. All the leads are taken through "Heyco" type grommets and these ensure a secure fastening. The PET lead is neatly equipped with a covered edge connector and will only fit one way round.

The RS232 lead is supplied with a standard "D" type (25 way) socket and again this is covered.

The only other obvious features of the unit are the two LED's and fuse holder. The LED's indicate "power on" and "listening."

The Heart Of The Matter

The circuitry of the interface is obviously UART based and the Baud rate clock is crystal controlled, but as some industrious persdon has sanded the IC numbers off the chips we can't tell you a lot more!

The Baud rate selection is done by using three of the switches on a four way DIP switch, the fourth being used for stop bit selection. The parity is set to EVEN.

Adjustment of the Baud rate is easy and using the code sheet provided in the instruction leaflet takes no time at all.

Program-A-Port

Five sample programs are listed in the instruction leaflet together with their expected printouts. To obtain a simple program listing via the adaptor the following instructions should be typed in:

```
OPEN 1 , 4
CMD 1
LIST
```

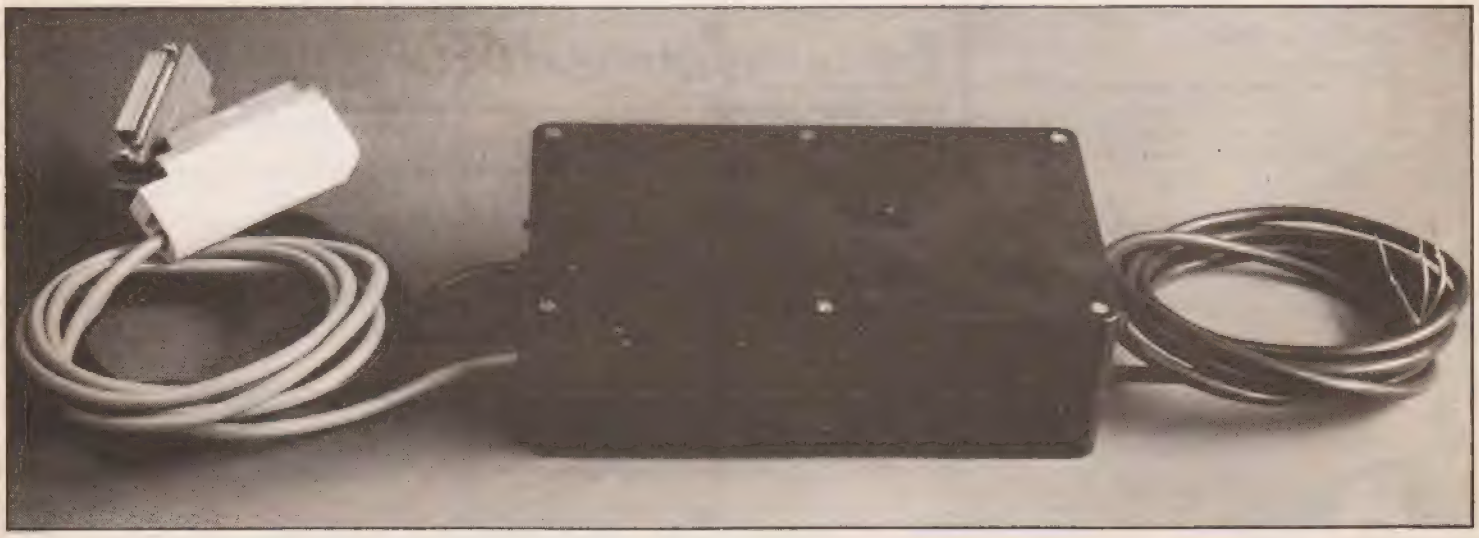
After typing CMD 1 the listening LED should be lit to indicate that the port has been called. When the listing is completed it is necessary to return the PET to the keyboard mode and this is done by creating a syntax error. This is easily done by typing a character on the keyboard followed by RETURN and the READY prompt will appear on the screen. The listen LED should go out at this point.

As the PET does not output actual space characters but relative tabs it is necessary to program the device to stop columns appearing all over the place and the adaptor has the necessary logic built in to overcome this problem.

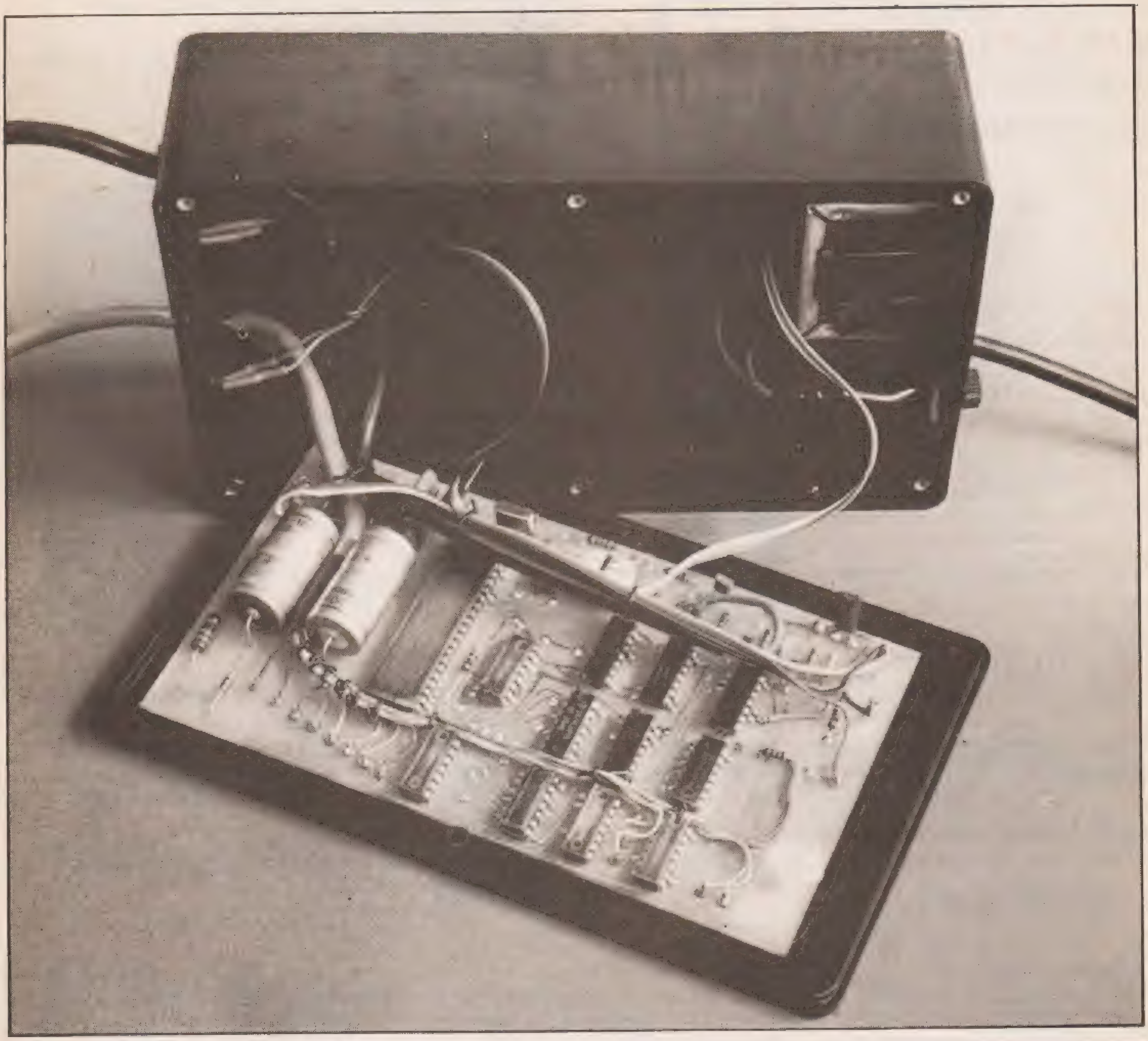
The programming examples contain useful little tips like this. If you require a previously written program to print instead of display you need only to type:

```
OPEN 1 , 4
CMD 1
RUN
```

and this will direct all the PRINT statements to the printer rather than to the screen. You may include the first two commands as part of your program but you will also have to insert the instruction CLOSE 1 , 4 at the end of the program.



PET PORT II



Summary

The adaptor fulfils all the requirements for the home and small business user. Without one it would not be readily possible to connect a printer to the PET. The Connecticut Word Processor program mentioned in our survey last month will work through this adaptor, but the supplied listing modifications will have to be inserted.

The unit costs £106 against £90 for the Petsoft unit.

The 3D unit is slightly neater in design and much easier to change Baud rates on, but apart from that they both perform exactly the same job.

As well a this RS232 adaptor 3D have a range of other PET peripherals including A to D converters and XY plotter interfacess and with luck we will be able to test some of these out at a latter date. 3D live at 43 Grafton Way, London W1P 5LA which is also the home of the London Computer Store.

```
10 REM PROGRAM A PRINTER TEST
14 OPEN1,4
15 PRINT#1,"PROGRAM A"
18 FORJ=1TO5
20 FORI=32TO126
40 PRINT#1,CHR$(I);:NEXTI
50 PRINT#1,:REM NEW LINE
60 NEXTJ
65 PRINT#1,:PRINT#1,
70 CLOSE1
95 REM
100 REM PROGRAM B FORMATTED OUTPUT WITHOUT
    CORRECTION
105 REM
110 OPEN1,4
```

```
115 PRINT#1,"PROGRAM B"
120 FOR I=1TO10
130 PRINT#1,I,SQR(I),I^2
140 NEXT
145 PRINT#1,:PRINT#1,
150 CLOSE1
195 REM
200 REM PROGRAM C CORRECTED FORMATTING
    USING RETURNS
205 REM
210 OPEN1,4
220 PRINT#1,"PROGRAM C"
230 FORI=1TO10
240 PRINT#1,I;CHR$(13)TAB(25)SQR(I);
    CHR$(13)TAB(50)I^2
250 NEXT
255 PRINT#1,:PRINT#1,
260 CLOSE1
295 REM
300 REM PROGRAM D CORRECTED FORMATTING
    USING LEN FUNCTIONS
305 REM
310 OPEN1,4
320 PRINT#1,"PROGRAM D"
330 FORI=1TO10
340 PRINT#1,ITAB(25-LEN(STR$(I)))SQR(I)
    TAB(25-LEN(STR$(SQR(I))))I^2
350 NEXT
355 PRINT#1,:PRINT#1,
360 CLOSE1
395 REM
400 REMPROGRAM E PRINTING ALTERNATELY ON
    PRINTER AND PET SCREEN
405 REM
420 OPEN1,4:OPEN 2,3
430 PRINT#1, "PROGRAM E PRINTER"
440 PRINT#2,"PROGRAM E SCREEN"
450 CLOSE1:CLOSE2
```

```
PROGRAM A
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz
 !"#$%&'()*+,-./0123456789:;<=>?@ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_`abcdefghijklmnopqrstuvwxyz


PROGRAM B
 1            1            1
 2            1.41421356           4
 3            1.73205081           9
 4            2            16
 5            2.23606798           25
 6            2.44948974           36
 7            2.64575131           49.0000001
 8            2.82842713           64
 9            3            81.0000001
 10            3.16227766           100


PROGRAM C
 1                        1                        1
 2                        1.41421356               4
 3                        1.73205081               9
 4                        2                        16
 5                        2.23606798               25
 6                        2.44948974               36
 7                        2.64575131               49.0000001
 8                        2.82842713               64
 9                        3                        81.0000001
 10                       3.16227766               100
```

Some Common BASIC Programs

76 short practical programs, most of which can be used on any microcomputer with any version of BASIC. Complete with program descriptions, listings, remarks and examples.

(200 pages)

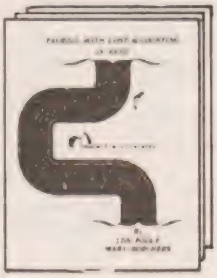
B1 **Price £6.00**

PROGRAMMING FOR LOGIC DESIGN

- **C 8080 Programming for Logic Design**
- **D 6800 Programming For Logic Design**
- **E Z80 Programming For Logic Design**

These books describe the meeting ground of programmers and logic designers; written for both, they provide detailed examples to illustrate effective usage of microprocessors in traditional digital applications.

B2/C/D/E (300 pages each)

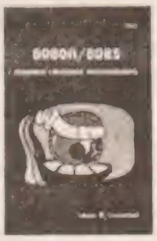


PROGRAM BOOKS WRITTEN IN BASIC

- **F Payroll With Cost Accounting** (400 pages)
- **G Accounts Payable And Accounts Receivable**
- **H General Ledger**

These books may be used independently, or implemented together as a complete accounting system. Each contains program listings, user's manual and thorough documentation. Written in an extended version of BASIC.

B3/F/G/H



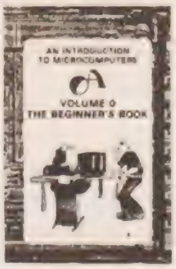
ASSEMBLY LANGUAGE PROGRAMMING

- **J 8080A/8085 Assembly Language Programming**
- **K 6800 Assembly Language Programming**

These books describe how to program a microcomputer using assembly language. They discuss classical programming techniques, and contain simplified proramming examples relevant to today's microcomputer applications.

B4/J/K (400 pages each)

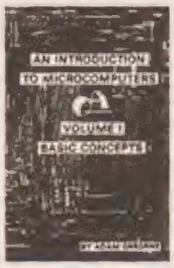
AN INTRODUCTION TO MICROCOMPUTERS



Volume 0 — The Beginner's Book

If you know nothing about computers, then this is the book for you. It introduces computer logic and terminology in language a beginner can understand. Computer software, hardware and component parts are described, and simple explanations are given for how they work. Text is supplemented with creative illustrations and numerous photographs. Volume 0 prepares the novice for Volume 1.

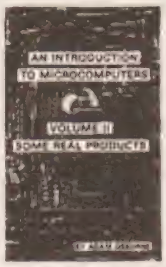
B5 (300 pages) **Price £5.65**



Volume 1 — Basic Concepts

This best selling text describes hardware and programming concepts common to all microprocessors. These concepts are explained clearly and thoroughly, beginning at an elementary level. Worldwide, Volume 1 has a greater yearly sales volume than any other computer text.

B6 (350 pages) **Price £6.00**



Volume II — Some Real Products (revised June 1977)

Every common microprocessor and all support devices are described. Only data sheets are copied from manufacturers. Major chip slice products are also discussed.

B7 (1250 pages) **Price £18.95** without binder

£24.40 with binder B4

HOW TO ORDER

Payment can be made by sending cheque, postal order, Barclay, Access or American Express card numbers. Write your name, address and the order details clearly, enclose 30p for post and packing or the amount stated. We do not wait to clear your cheque before sending the goods so this will not delay delivery. All products carry 1 year guarantee and full money back 10 day reassurance. All prices include VAT.

Trade enquiries: Send for a complete list of trade prices — minimum order value £100.
Telephone Orders: Credit card customers can telephone orders direct to Daventry or Edgware Road.
24 hour phone service at both shops: 01-723 4753 03272-76545.

CALLERS WELCOME
Shops open 9.30 — 6.00.

Metac

ELECTRONICS & TIME CENTRES

North & Midlands
67 High Street, DAVENTRY
Northamptonshire
Telephone: 03272 76545

South of England
327 Edgware Road
LONDON W.2
Telephone: (01) 723 4753

NASCOM 1 KIT or BUILT

SUPERBOARD II ON DEMO NOW

2K BASIC FOR STANDARD NASCOM 1

ELECTRONIC MASTERMIND NEW

Play 9 Mastermind games against the computer or set your own secret code 3, 4 or 5 digits.

£14.90 Free mini Mastermind with each order

CHESS CHAMPION
6-Level Chess Computer NEW

£89.50 with mains adaptor

STAR CHESS T/V GAME
in colour.
A Spage Age chess game NEW

£59.50 with mains adaptor

CHROMA-CHIME
24 Tune door chimes

kit **£9.95** built **£14.95**

TRS 80 — 100 Programs on Tape

£46.50 SAE with all enquiries

Buy it with Access

N.I.C. MODELS

Please mention CT when contacting advertisers

Dear Computing Today,

It was nice to see, at last, someone printing a program for NASCOM users. And the information on getting the Monitor to load during a program works rather well. But.....

It would have been rather good if the listing had been in Z80 Assembly mnemonics, and correct. It took me ages to sort out, and I eventually gave up and wrote a better program. So there!

Examples so that you know I'm not bull-6502-ing. Line 0F65, 10 FB is DJNZ -3 which doesn't get you to Line 1.
Line 0F79, 10 ED is not JR NZ, it's DJNZ.

I'd bet my version against an 8K RAM board that it wasn't assembled on a NASCOM.

To change the subject, the magazine I like but what happened to the review of NASCOM add-ons? I know, they haven't been delivered, which is absolutely typical.

People who deliver:–
Microdigital (Superb service)
People who don't:–
NASCOM, Science of Cambridge, Comp Computer Components.

If you ordered from the last two you are going to have to wait until we get ours.

Yours mnemonically,
Chris Blackmore.

31 Herne Rise,
Ilminster,
Somerset TA19 0HH

Dear Sir,

While not wishing to be too critical I feel that I must agree with the letter from Mr Anderson in your April issue. The photographs you use are often of very poor quality, in fact there is a case in point directly above his letter on page 22. Those accompanying the WCE show report were little better, the picture at the top of page 60 looking as though it's been double exposed on an out-of-focus instamatic.

I do have one other criticsm. That is concerning the quality of grammar and spelling in some of the articles. I can see many of the same faults in your 'parent' magazine, ETI.

As for the bias towards TRITON, this is undeniable but then every computer magazine has its favourite systems.

I also concur with Mr Anderson on the BASIC feature and his comments on Donald Alcocks splendid little book. It has all been done before. It should be the job of a magazine to lead rather than follow.

I hope that you don't consider the above comments too destructive. I look forward to seeing CT improve with age– a healthy computer press is important in increasing awareness and stimulating interest in the field.

Yours sincerely,
D.Burns

11 Turpins Chase,
Oaklands,
Welwyn,
Hertfordshire.

Dear Sir,

I am very grateful for all the correspodence, and pleased to see that a NASCOM program was long awaited by mant of your readers. I regret that three typographical errors occurred, although many of you found them for yourselves. The corrections are as follows:–

Address 0DDC should read 0B and not 0A,
Address 0EE8 should read 00 and not FA,
Address 0F6D should read 1E and not 13.

Whilst one or two mnemonics were misprinted these are too trivial to be of consequence. Whilst every attempt is made to ensure an accurate listing it is not humanly possible to quarantee perfection, and the user must be prepared to do some debugging. For those of you that met with success I hope you find it useful.

One or two people have been perturbed by not automating their cassettes. For them I suggest loading the program and registers each time from their cassette and executing from 0C77. This will bypass the auto load facility.

For the benifit of Mr Blackmore, who I hope is sending me his program, the design of 'Ledger' was carried out and is being currently used on my NASCOM.

Yours faithfully,
M.J.Bell T.Eng (CEI) MITE.

Dear Sir,

With reference to the 25120 WOM chips (Data Sheet, April Issue) I have great pleasure in informing you that they fulfil admirably my requirements for a three cycle confabulator.

I am at present designing a Mark II of ERIS, Electronic Reality Identification Shifter, which will utilise reverse-biased pseudo-Rabbet E.D. with Zen diodes linked via' involatile CPT-symmetric lepton shunts functioning as 'pseudo-neurons'. This will enable data for the confabulator to be WOMed into the cerebronic parts of the circuit without Deere-Lyza bucket loss occuring.

A few problems though:–

1) The SEX process. Several of the chips purchased were found to have a considerably deviant bias, necessitating much use of M to M connectors and a twisted pair or two.

2) The non hermetic sealing. Since ERIS needs completely immutable environs, I have searched far and wide for a Seal of Hermes. Will Solomons Seal, which grows nearby, do instead?

3) The cooling problem. I have had no end of problem in finding a six foot fan, so being a do-it-oneself type, I persuaded a friend to crochet me one in Raffia. This is unsatisfactory as either,
a) his arms become EXTREMELY tired, giving functional periods of less than five seconds before burn out, or
b) he becomes confabulated by the local field (½" is a bit near!!) and thrashes the setup with the fan.

Yours faithfully,
Magenta Screens (Ms)

56 Frederick Street,
Loughborough,
Leics. LE11 3BJ

Mr. Ron Harris
Acting Editor
Computing Today
25-27 Oxford St.
London W1R 1RF

Dear Mr Harris,

I read with interest the article on Word Processors in the recent May issue. In my company we have two TRS-80 Systems , one of which has been slightly modified to run the word processing package on which this letter was produced.

This is NOT the package apparantly supplied by A.J. Harding and I can only express surprise that your article should give the impression that his package is the only one available for the TRS-80. Our package is the 'Electric Pencil' produced by Micheal Shrayer Software and distributed in the U.K. by a number of TRS-80 specialists.

THe 'Electric Pencil is available for both the cassette tape and disc systems at a cost of around £100 which includes both software and a keyboard modification to produce lower case.

The system provides a screen orientated input and editing process. Files may be saved,loaded or merged. There are also facilities for:-

screen scrolling forwards/backwards at variable speeds.
simple search or search with replace.
text block moves/deletes
line insert/delete
automatic heading with page numbering
left and right justification
line length control
variable print speeds e.g. 300 or 1200 Baud

In terms of capacity we have found that the system can handle between 20 to 30 pages of A4 text.

The system is almost bug free and the documentation is beyond complaint. In fact it is necessary to look at very up market system before any real limitations appear e.g.:-

No true proportional output
No centering or multiple column facilities (not really a limitation with a screen orientated system)

109 King Charles Rd.
Surbiton,
Surrey
May 9th 1979

Your sincerely

Derrick Rowe

Mr. P. Wickett

Minefield Game

The program that follows was designed specifically for the Commodore PET computer. It uses less than 3K bytes of memory for storage and execution but produces a very effective and compelling game. The object is to save two "victims" from a minefield. However, the mines increase in number each time you move and you must watch where you are moving or else you will be blown up!! There is a time limit of one minute (this can easily be changed) at the end of which you will be destroyed unless you have rescued both victims.

```
1   PRINT "  DO YOU WANT THE RULES (Y OR N)"
2   INPUT Q$: IF Q$ = "Y" THEN 2000
3   PRINT "  SET NO. OF MINES PER MOVE (10-30)
4   INPUT V:PRINT "   "
5   IF V > 30 THEN 3
6   IF V < 10 THEN 3
7   FF=INT (RND(1)*999.9)+32768
8   JJ=INT(RND(1)*999.9)+32768
9   POKE FF,87;POKE JJ,87
10  F=33268
12  POKE F,81
13  X=102
14  LET TIME $ = "000000"
16  FOR H=1 TO V
17  D=INT(RND(1)*999.9)+32768
18  IF D=FF THEN 17
20  IF D=JJ THEN 17
24  IF PEEK(D)=X THEN 17
26  IF D=F THEN 17
28  POKE D,X
30  NEXT H
40  IF TIME $ 7 "000/00" THEN 200
50  GET C$:IF C$ ""THEN 50
51  IF C$ ="I" THEN 520
52  IF C$ ="2" THEN 530
53  IF C$ ="3" THEN 540
54  IF C$ ="4" THEN 550
56  IF C$ ="6" THEN 560
57  IF C$ ="7" THEN 570
58  IF C$ ="8" THEN 580
59  IF C$ ="9" THEN 590
60  IF C$ ="R" THEN 505
100 GOTO 50
200 PRINT "SORRY YOU HAVE RUN  OUT OF TIME"
200 PRINT "SORRY YOU HAVE RUN OUT OF TIME"
499 FOR BB=1 TO 400:NEXT BB
500 POKE F,42:POKE F+I,64:POKE F-I,64
501 POKE F-40,66:POKE F+40,66
502 POKE F-41,77:POKE F-39,78:POKE F+39,78
503 POKE F+41,77
504 FORK BB=1 to 2000:NEXT BB
505 PRINT "  ANOTHER GAME(Y OR N)?"
506 GET P$:IF P$="" THEN 506
508 IF P$= "Y" THEN 3
509 IF P$= "N" THEN 515
510 GOTO 506
515 STOP
520 A=39
525 GOTO 620
530 A= 40
535 GOTO 620
540 A= 41
545 GOTO 620
550 A= -1
555 GOTO 620
560 A= +1
565 GOTO 620
570 A= -41
575 GOTO 620
580 A= -40
585 GOTO 620
590 A= -39
620 POKE F,46
621 F=F+A
622 IF PEEK(F)=X THEN 499
623 POKE F,81
624 IF PEEK(FF)<> 87 THEN 626
625 GOTO 16
626 IF PEEK (JJ)<>87 THEN 1000
628 GOTO 16
1000 IF TIME$ > "000100" THEN 200
1004 PRINT " CONGRATULATIONS YOU HAVE "
1010 PRINT " SAVED THE VICTIMS FROM THE "
1015 PRINT " MINEFIELD WITHOUT GETTING "
1020 PRINT " KILLED. WOULD YOU LIKE TO "
1025 PRINT " TRY AGAIN(Y OR N)? "
1030 GET YS:IF YS= "" THEN 1030
1040 IF YS = "Y" THEN 3
1044 IF YS = "N" THEN 1050
```

SOFTSPOT

```
1048  GOTO 1030
1050  STOP
2000  PRINT " YOU ARE THE '●' SYMBOL
      IN THE "
2004  PRINT " MIDDLE  OF THE SCREEN.
      YOU MUST "
2008  PRINT " TRY AND REACH THE 2 'O'
      SYMBOLS "
2010  PRINT " AND MOVE OVER THEM TO
      PROTECT THEM "
2012  PRINT " BUT WATCH  OUT FOR THE MINES, "
2014  PRINT " THEY WILL DESTROY YOU IF YOU "
2016  PRINT " MOVE INTO  THEM.  YOU MAY
      MOVE "
2018  PRINT " YOURSELF USING KEYS 1-9. "
2020  PRINT " THERE IS A TIME LIMIT OF "
2022  PRINT " 1 MINUTE. PUSH R TO START "
2024  PRINT " AND STOP..GOOD LUCK!! "
2030  GET M$:IF M$ = ""THEN 2030
2040  IF M$  = "R" THEN 3
2050  GOTO 2030
```

computing today

technical book service

What Is A Microprocessor ?
2 Cassette tapes plus a 72 page book deal with many aspects of microprocessors including Binary and Hexadecimal counting, Pro— gramming etc. £10.75.

NEW:
Adams, C. BEGINNERS GUIDE TO COMPUTERS AND MICROPROCESSORS WITH PROJECTS £5.60
Understanding building programming and operating your own microcomputer.

NEW:
Albrecht, B. BASIC FOR HOME COMPUTERS. A self teaching guide £4.75
Shows you how to read, write and understand basic programming language used in the new personal size microcomputers.

Albrecht B. BASIC. A self teaching guide (2nd edition) £4.50
Teach yourself the programming language BASIC. You will learn how to use the computer as a tool in home or office and you will need no special maths or science background.

Alcock, D. ILLUSTRATING BASIC £2.60
This book presents a popular and widely available language called BASIC, and explains how to write simple programs.

Altman, I. MICROPROCESSORS £10.65
Gives a general overview of the technology design ideas and explains practical applications.

Altman, L. APPLYING MICROPROCESSORS £12.00
Follow volume which takes you into the second and third generation devices.

Aspinall, D. INTRO TO MICROPROCESSORS £6.40
Explains the characteristics of the component.

NEW:
Barden, W. Z-80 MICROCOMPUTER HANDBOOK £7.65

Barden, W. HOW TO BUY AND USE MINICOMPUTERS AND MICROCOMPUTERS £7.75
Discusses these smaller computers and shows how they can be used in a variety of practical and recreational tasks in the home or business.

Barden, W. HOW TO PROGRAM MICROCOMPUTERS £7.00
This book explains assembly language programming of microcomputers based on the Intel 8080, Motorola MC6800 and MOS Technology MCS6502 microprocessor.

Barna, A. INTRODUCTION TO MICROCOMPUTERS AND MICROPROCESSORS £7.50
Provides the basic knowledge required to understand microprocessor systems. Presents a fundamental discussion of many topics in both hardware and software.

Bibbero, R. J. MICROPROCESSORS IN INSTRUMENTS AND CONTROL £12.45
Introduces the background elements, paying particular regard to the dynamics and computational instrumentation required to accomplish real-time data processing tasks.

Lancaster, D. TV TYPEWRITER COOKBOOK £7.75
An in-depth coverage of tv typewriters (tv's) the only truly low cost microcomputer and small display interface.

Lancaster, D. CHEAP VIDEO COOKBOOK £6.50

Lesea, A. MICROPROCESSOR INTERFACING TECHNIQUES £7.50

NEW:
Leventhal. INTRO TO MICROPROCESSORS £16.70

NEW:
Lewis, T. G. MIND APPLIANCE HOME COMPUTER APPLICATIONS £4.75

NEW:
Libes, S. SMALL COMPUTER SYSTEMS HANDBOOK £5.75
The Primer written for those new to the field of personal home computers.

NEW:
Lippiatt. ARCHITECTURE OF SMALL COMPUTER SYSTEMS £4.35

NEW:
Moody, R. FIRST BOOK OF MICROCOMPUTERS £3.85
(the home computer owners best friend).

McGlynn, D. R. MICROPROCESSORS — Technology, Architecture & Applications £8.40
This introduction to the 'computer-on-a-chip' provides a clear explanation of the important new device.

McMurran, PROGRAMMING MICROPROCESSORS £5.50
A practical programming guide that includes architecture, arithmetic/logic operations, fixed and floating point computations, data exchange with peripheral devices computers and other programming aids.

NEW:
Nagin, P. BASIC WITH STYLE £4.00
Programming Proverbs. Principles of good programming with numerous examples to improve programming style and producing.

NEW:
Ogdin SOFTWARE DESIGN FOR MICROCOMPUTERS £7.00

NEW:
Ogdin. MICROCOMPUTER DESIGN £7.05

Peatman, J. B. MICROCOMPUTER BASED DESIGN £19.00

This book is intended for undergraduate courses on microprocessors.

NEW:

Bursky, D. MICROCOMPUTER BOARD DATA MANUAL £5.40

Bursky, D. MICROPROCESSOR DATA £5.40

Includes complete description of the processor. Support circuits, Architecture, Software, etc.

Coan, J. S. BASIC BASIC £7.50

An introduction to computer programming in BASIC language.

Coan, J. S. ADVANCED BASIC £7.30

Applications and problems.

NEW:

Duncan. MICROPROCESSOR SOFTWARE ENGINEERING £13.50

NEW:

Freiberger, S. CONSUMERS GUIDE TO PERSONAL COMPUTING AND MICROCOMPUTERS £5.50

NEW:

Frenzel, L. GETTING ACQUAINTED WITH MICROPROCESSORS £7.10

This is an invaluable book for those who want to know more about hobby and personal computing.

Gilmore, C. M. BEGINNERS GUIDE TO MICROPROCESSORS £4.75

Gosling, R. E. BEGINNING BASIC £3.25

Introduces BASIC to first time users.

Graham, N. MICROPROCESSOR PROGRAMMING FOR COMPUTER HOBBYISTS £7.00

Haviland, N. P. THE COMPULATOR BOOK £6.20

Building super calculators and minicomputer hardware with calculator chips.

Heiserman, D. L. MINIPROCESSORS FROM CALCULATORS TO COMPUTERS £4.85

Hilburn, J. L. MICROCOMPUTERS, MICROPROCESSORS, HARDWARE, SOFTWARE AND APPLICATIONS £16.95

Complete and practical introduction to the design, programming operation, uses and maintenance of modern microprocessors, their integrated circuits and other components.

Klingman, E. MICROPROCESSOR SYSTEMS DESIGN £16.95

Outstanding for its information on real microprocessors, this text is both an introduction and a detailed information source treating over a dozen processors, including new third generation devices. No prior knowledge of microprocessors or microelectronics is required for the reader.

Kemeny, J. G. BASIC PROGRAMMING £6.10

A basic text.

Korn, G. A. MICROPROCESSOR AND SMALL DIGITAL COMPUTER SYSTEMS FOR ENGINEERS AND SCIENTISTS £19.00

This book covers the types, languages, design software and applications of microprocessors.

Rao, G. U. MICROPROCESSOR AND MICROPROCESSOR SYSTEMS £20.50

A completely up-to-date report on the state-of-the-art of microprocessors and microcomputers written by one of the leading experts.

Rony, P. H. THE 8080A BUGBOOK: Microcomputer Interfacing & Programming £8.15

The principles, concepts and applications of an 8-bit microcomputer based on the 8080 microprocessor IU chip. The emphasis is on a computer as a controller.

Scelbi. 6800 SOFTWARE GOURMET GUIDE AND COOKBOOK £7.80

Scelbi. 8080 SOFTWARE GOURMET GUIDE AND COOKBOOK £7.80

Scelbi. UNDERSTANDING MICROCOMPUTERS £7.60

Gives the fundamental concepts of virtually all microcomputers.

NEW:

Schoman, K. THE BASIC WORKBOOK £3.70

Creative techniques for beginning programmers.

NEW:

Sirion, D. BASIC FROM THE GROUND UP £6.00

Soucek, B. MICROPROCESSORS AND MICROCOMPUTERS £18.80

Here is a description of the applications programming and interfacing techniques common to all microprocessors.

NEW:

Spracklen, D. SARGON £9.75

A computer chess program in Z-80 assembly language.

NEW:

Tracton. 57 PRACTICAL PROGRAMS & GAMES IN BASIC £6.40

Programs for everything from Space war games to Blackjack.

Waite. M. MICROCOMPUTER PRIMER £6.25

Introduces the beginner to the basic principles of the microcomputer.

Ward. MICROPROCESSOR / MICROPROGRAMMING HANDBOOK £6.00

Authoritative practical guide to microprocessor construction programming and applications.

NEW:

Veronis. MICROPROCESSOR £12.85

Zaks, R. INTRODUCTION TO PERSONAL AND BUSINESS COMPUTING £7.50

Zaks, R. MICROPROCESSORS FROM CHIPS TO SYSTEMS £7.50

Note that all prices include postage and packing. Please make cheques, etc, payable to Computing Today Book Service (Payment in U.K. currency only please) and send to:
Computing Today Book Service,
P.O. Box 79,
Maidenhead, Berks.

Mr. I. Graham

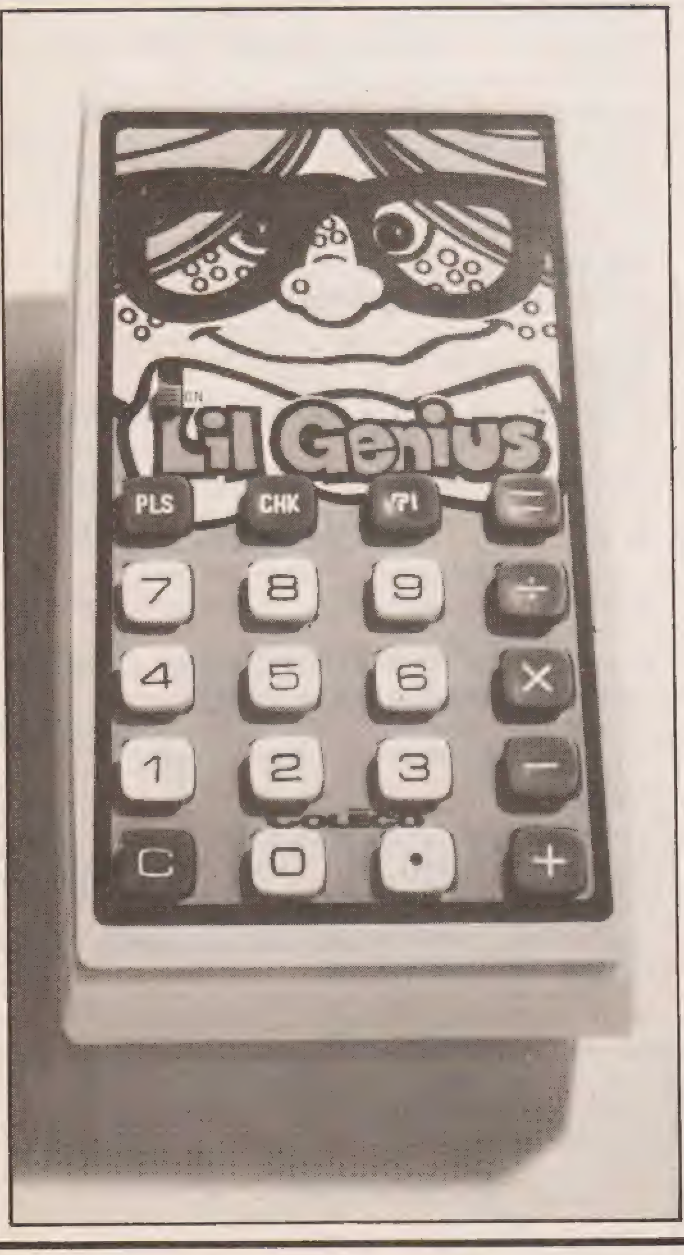
A survey of the electronic games that are available on the market

Last month we presented a survey of microcomputers. This month we decided to turn our attention to electronic games. It was a good excuse to get a few games in and play with them for a while.

However, as the games began to take over the office, we wondered who was the boss. Didn't somebody once say, 'Whom the Gods would destroy they first give electronic games?' Well, it was something like that!

We looked mainly at microprocessor-based, hand-held games and board games. It seems that if you want to play draughts or chess, break codes or blast flying saucers out of existence, there's a game to do it.

A north London electronics firm is even marketing a



Lil Genius

If your maths is a little rusty, Lil Genius may be able to help you out. Together with the short manual provided, Lil Genius is designed to help teach the rudiments of arithmetic to children of five and above.

Having entered what you believe to be the correct answer to a calculation, you can ask Lil Genius if you are correct by pressing three keys in turn. If you're wrong, a freckle-faced bespectacled youngster adorning the front panel winks a nasty case of red eye (for our optician readers) at you and a bleeper sounds. In other words a red LED flashes on and off if you're wrong.

If you manage to get your sums right, the freckly-face winks a green eye at you . . . but no bleeping,no sound at all. Frankly, we found it more entertaining to lose.

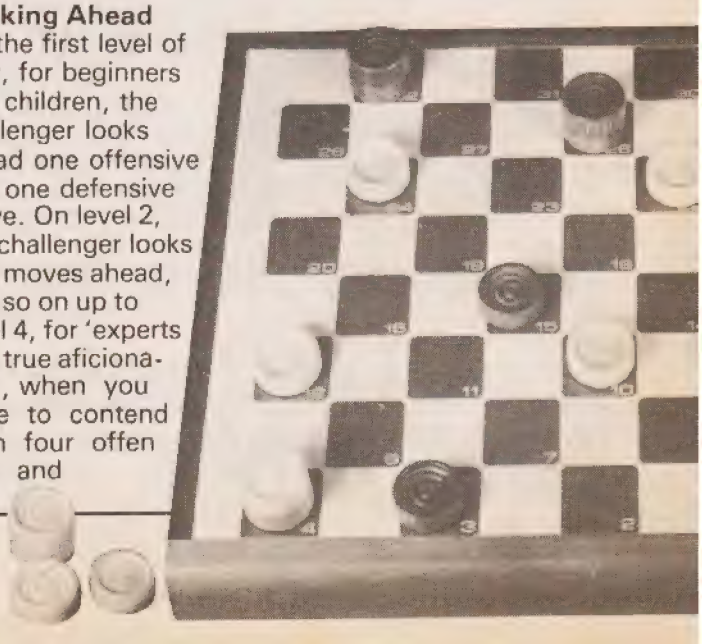
Lil Genius will cross the counter at a very reasonable £6.45.

Checker

Looking very similar to Chess Challenger, Checker Challenger 4 is an attractively presented, four level, microprocessor-based draughts game.

Looking Ahead

On the first level of play, for beginners and children, the challenger looks ahead one offensive and one defensive move. On level 2, the challenger looks two moves ahead, and so on up to level 4, for 'experts and true aficiona-dos', when you have to contend with four offen sive and



GAMES SURVEY

baseball game. If you want to play soccer in your hand, however, you'll have to go to Aberdeen. No, not the well-known jumping off point for the nearest oil rig. This particular box of tricks hails from Aberdeen, Hong Kong.

We were particularly interested to see a new range of hand-held computer games from America, introduced to this country by Spectrum Marketing. Most are appearing on the British market now and others will do so within the next month or two.

Predictably we found that, in some cases, well used sound effects could turn the boring process of button pushing into an entertaining game. The undoubted winner of our Golden Digit award for the game with the mostest is UFO Master Blaster, which must be set to be a best seller in the next year. Although a little pricey for one game at £21.95, we feel that it is worth every penny of it. At times there was a queue in the office, of people wanting to play with it next and certain members of staff were found guilty of queue-jumping. Well, we're all kids at heart, aren't we?

Digits

Although the name Digits may not be familiar to you, the game certainly will be. The cunning little devil thinks up a four digit code and you have to crack it. You are given clues to tell you how close you are to the right answer. You are told how many digits you have correct and in the right position and correct but in the wrong position.

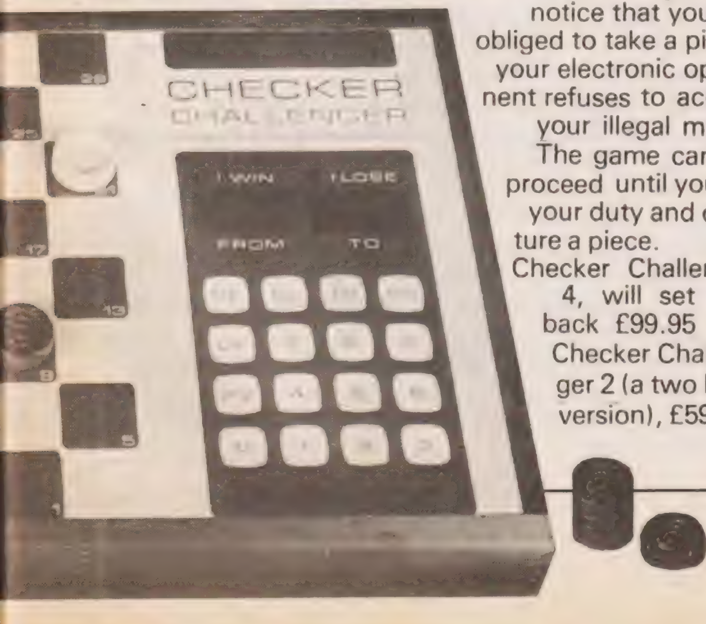
This was the only game from Spectrum where we found it adviseable to read the instruction booklet on, before launching into a game. During one game we found that some digits would not appear on the display when their buttons were pressed. A broken game? A swift scan of the instruction booklet revealed that only digits of five or less make up the code on the lowest of the two skill levels. If you dare switch to level two, you'll have to cope with the mind-boggling choice of 0 to 9. Fun, while the novelty lasts. Digits is expected to retail at £13.95.

Challenger

defensive moves ahead. However, on level four excessive response times tend to make games seem interminable.

Do Your Duty If you try to cheat by choosing not to notice that you are obliged to take a piece, your electronic opponent refuses to accept your illegal move. The game cannot proceed until you do your duty and capture a piece.

Checker Challenger 4, will set you back £99.95 and Checker Challenger 2 (a two level version), £59.95.



Star Chess

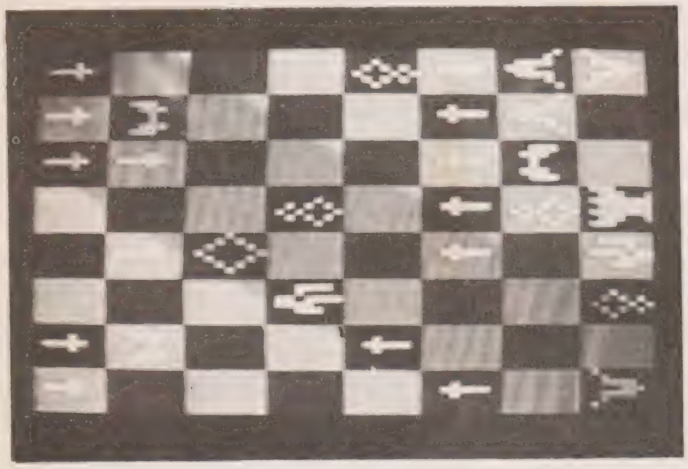
Star Chess is a remarkable new game from Videomaster. Plug it into the aerial socket of a domestic colour telly and you'll see a crystal clear, beautifully sharp, full colour chess board with a few very strange pieces on it.

As the name suggests, it has something to do with chess, so a basic knowledge of the game is a must. However, if you think this is just another TV chess game, you're completely wrong.

Each piece is moved by means of a cursor, controlled by four buttons on each of two hand controllers. Every movement is accompanied by bleeps and squeaks from the built-in sound unit. The pieces, with the exception of pawns, move as in chess. Pawns cannot take diagonally but they can move sideways and backwards.

Firepower

What puts this game in a class of it's own is the ability of pieces to fire missiles at each other. Yes, that's what I



Master Mind

Zodiac

This first ever astrology computer is claimed by Spectrum to be 'the perfect answer for all those people who believe their lives are influenced by the stars.'

When used with the manual provided, the computer astrologer gives you three types of information. In the horoscope mode, a complete horoscope can be built up for anyone, including planetary positions at the time of birth and corresponding personality characteristics.

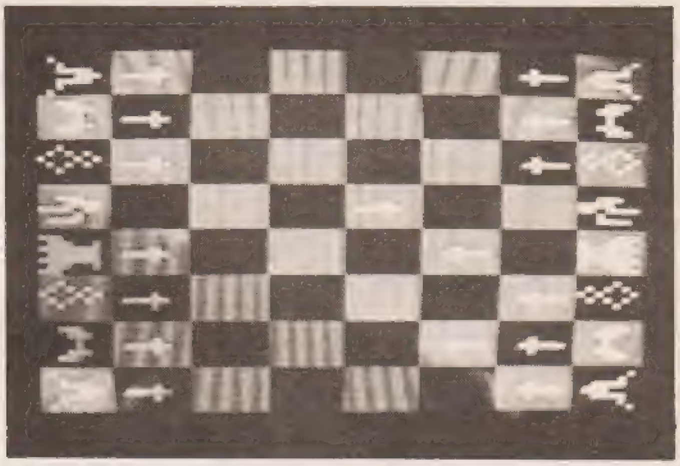
On daily preview, Zodiac will suggest what you should do from day to day, according to how the stars appear for that day — past, present and future.

In the advice mode, Zodiac will offer advice on any course of action entered by the worried star gazer. (Is it safe to feed the budgy today, I wonder?)

If your crystal ball has seen better days, why not put it out to grass and make some room for the Zodiac Astrology Computer, which you can get your hands on for around £25.95.

said, they can fire missiles at each other. If taking your opponent's piece endangers yours, then keep your distance and fire a missile at it instead. A hit is not guaranteed! However, if you do score a direct hit, you destroy some of your opponent's shields, which can number from two (pawn) to seven for the King and Queen. Ammunition is similarly limited.
Now, you won't believe this. How many times have you wished that you could forget about the rules of chess and move a piece to somewhere else on the board to get it out of trouble, or take it off the board completely for a while. You can do just that with this game. You can 'warp' a piece off the board. The drawback is that it can reappear anywhere, at any time. It can even land on top of one of your own pieces, which it then replaces.
Your objective is to destroy your opponents King. Whilst it cannot be taken on it's own starbase (home square), it can be fired at.
This is undoubtedly the most absorbing TV game ever to darken our doorstep and it retails (the game, not the doorstep) at £59.95.

A familiar name for a familiar game. This is the electronic version of the well-known board game played with coloured pegs.
In this case the colour code is replaced by a four digit number. The display shows you if your attempt to break the code set by the machine has any correct digits in the wrong place, or correct digits in the right place. A tally sheet is also provided, so that you can keep a record of your unsuccessful attempts.
This game allows you to either play solo, or against another player, who can use the set function to give you a code to break. If you are getting nowhere and give up, you can see the code you were aiming for by pressing the FAIL button. If, however, you are successful, the game will display the number of attempts you made on the way to the correct solution.
We found a way of cheating this unit and your real opponent into thinking that you can break the toughest code first time, every time. See if you can work it out. Remember, you can see what the answer is by pushing the FAIL button. Mastermind is made by Invicta.



Simon

Simon, from Milton Bradley, is a game of memory, incorporating three variations and four skill levels, an electronic version of the 'Simon Says' game.

Raspberry

The idea is to repeat the ever-increasing random light flashes that Simon generates, creating a longer and longer sequence. On the highest skill level, can your memory stretch to a 31 lamp sequence? A wrong move illicits a disdainful raspberry from Simon.

TI Chip

The game is based on Texas Instruments' TMS1000 microprocessor. A set program is stored in ROM, while a RAM takes care of player-entered information.
Although already very successful in America, Simon was only recently launched on to the British market at a lavish party hosted by Elaine Stritch and Donald Sinden.
For this electronic application of a simple idea, you will have to pay around £29.00.

Master Blaster

This is undoubtedly the best of the new range from Spectrum, to the point of being addictive. We think it should be sold with a health warning.

The principle of the game is to kill or be killed. You live at one end of the display and, at random time intervals, you are attacked by flying saucers coming from the other end of the display. The attackers can fly along one of three tracks towards you and can switch between tracks in mid-attack. To complicate matters, two saucers can attack simultaneously, possibly changing flight paths on their way.

Your job is to blast them off the display in order to stay alive. To enable you to do just that with your single missile, you have two controls — a FIRE button, which is self-explanatory, and a three position, centre-biased switch with which to select your flight path.

As you knock off the opposition, you build up a score. The idea is to gain a total score of 99 to win. The length of the display is graduated from one, at your home base, to six at the far end. The number opposite to where you wipe out an enemy saucer is your score for the engagement. So, the further away from home the kill is, the more you get for it. It pays, therefore, to try to anticipate when the enemy will appear.

As your skill increases you can step up the speed of the game with a three position switch, from an easily manageable beginner's speed to an infuriating, almost impossible blur.

The game is complex, but not intricately so. You don't have to pour over a long book of instructions before you can play. It is attractively presented in an unusually shaped case, which fits naturally and comfortably into the hands, allowing easy access to the controls. There is also the added attraction of sound effects, without which the game might be merely so many flashing lights on the display.

Like a good book, this game is impossible to put down. The speed selection, saucers changing flight paths, the possibility of scoring more by anticipating attacks, sound effects and the randomness of attacks all go together to make this an immensely enjoyable game to play.

Priced at £21.95 the UFO Master Blaster is destined to be a winner.

Amaze-A-Tron

This robust package boasts eight games in one. The Amaze-A-Tron maze game consists of a playing field keyboard on which players attempt to move from one point to another, both starting and finishing points being given by the unit at the outset, via a secret maze path. The path is traced out on keyboard squares numbered 1 to 25, with over a million maze path variations.

There are six two player (competitive) games and two solo games. When the unit is switched on, P1 (program 1) appears in a small display on the front panel. This is the game number. Anything from P1 through to P8 can be selected. A panel on the back of the game gives brief details of the eight games — a prudent insurance against instruction leaflet loss. The various games manage to incorporate blind turns, cul-de-sacs and false routes.

Zap

Skill is to this game what rice pudding is to rugby union. Spectrum describe this as a 'Computerised electronic 'missile' game for two players designed to test hand and eye co-ordination to their limit.'

What does it do? The LEDs along the serpent's back flash in turn, giving the impression of a light travelling from one end of the game to the other. The idea is to stop the light from reaching your end of the game by stabbing at your control button when the light turns towards you. Your opponent has the same thought in mind. The light speeds up as it travels from end to end.

When you finally succumb to fatigue or boredom and the light slips through a LED on the serpent's back lights to show you the score before the next round.

For this electronic ping-pong, you'll pay around £10.95.

Then the start button can be pressed. The display first shows the start number of the maze. When that is pressed and the green marker is put in position, the display will show your destination, the finish number, on which the red marker is placed. When this is pressed, the game begins.

The unit begins to tick. If you don't make your move in 10 seconds the game self destructs, well almost. Assuming you hit a number within the 10 second limit, how do you know whether or not it's correct? Sound effects to the rescue. Hit the wrong number and the game blows a raspberry at you — charming. However, a happier melody celebrates your correct number choice. If you are playing a two player game, red and green LEDs indicate whose turn it is, with separate correct number choice tunes. When you reach your destination successfully, a winner's song announces your arrival.

Sound effects make this game a joy to play. Amaze-A-tron has a recommended retail price of £17.95.

Chess Challengers

Available since April, Chess Challenger 7 is a seven level version of the computerised chess board which pioneered the UK market for microprocessor games. This miracle machine will either play against you or against itself. It allows you to change sides in mid-game, on any move.

Now available is 'son of Chess Challenger 7', namely Chess Challenger 10. You've guessed it — ten levels of play. Since the pieces are magnetised, you can turn the board upside down and not one piece will fall off its simulated leather or brushed gold foil square — for the Australian export market, do you think?

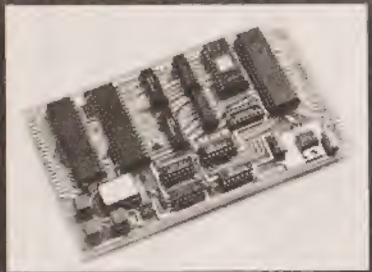
If you should execute a move incorrectly, ie misread the move shown on the display and move that piece incorrectly, it's not long before you find out about it. A few moves later, Chess Challenger will refuse to accept what appears to be a perfectly legitimate move, because its memory tells it where pieces should be, not where they are. Fortunately the player has access to the super memory. You can ask Challenger to show you, piece by piece, where everything is.

From July you will be able to buy Voice Chess Challenger — the first of this variety of solo chess game in the world that speaks. It is claimed to be the ultimate in microprocessor mini-computer wizardry.

Voice Chess Challenger features the strongest chess program ever placed in a microprocessor and is approximately twice as fast as other models in the series — that has to be an improvement at the higher levels of play. In fact this Challenger has 'infinite' levels of play. It will continue to compute its move until the halt key is pressed. The program has a large repertoire of chess book openings. At the end of the game it displays the number of moves played. The big difference between this 'wizard' and its progenitors is that it speaks every move and capture and will repeat board positions on demand.

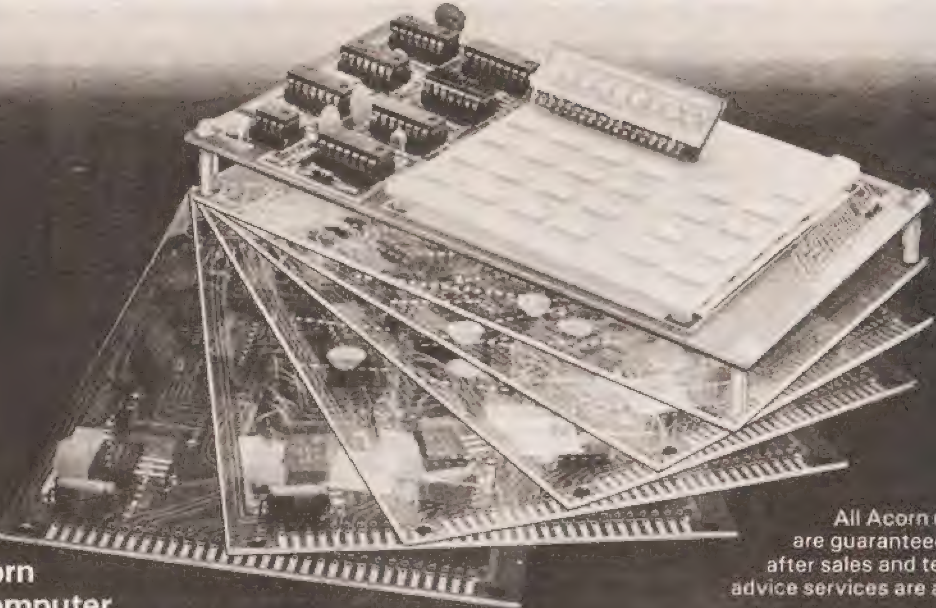
Chess Challenger 7 is the first in this series to retail at under £100. Chess Challenger 10 will burn a £200 hole in your pocket, while Voice Chess Challenger will sell at an understandable £250.

Three Trumps from Acorn



Acorn Controller

Designed as an industrial controller module, it is based on the 6502 CPU with 2K Eprom, 1.25K ram and 32 I/O lines. In eurocard format it is provided with an onboard monitor (2 × 74S571) giving comprehensive development and debugging facilities. Also available in minimum configuration for low cost OEM applications.

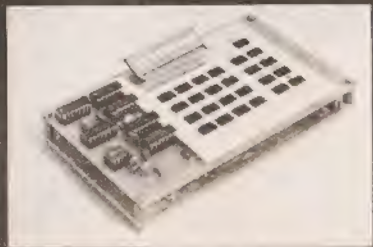


The Acorn Microcomputer

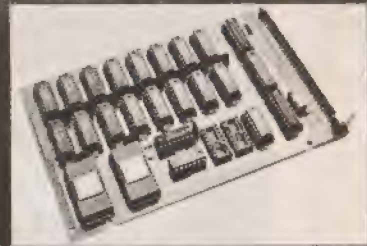
The Acorn controller module mounted beneath a matching eurocard with hex keyboard, 8 digit seven segment display and CUTS tape interface requires only a single unstabilised power supply to form the powerful Acorn microcomputer.

Although designed for expandability the Acorn Microcomputer is a complete development system for the Acorn controller and together with the Acorn Users Manual provides the perfect introduction to hex programming; the carefully optimised monitor has the following functions:

System Program
Set of sub-routines for use in programming
Powerful de-bugging facility displays all internal registers
Tape load and store

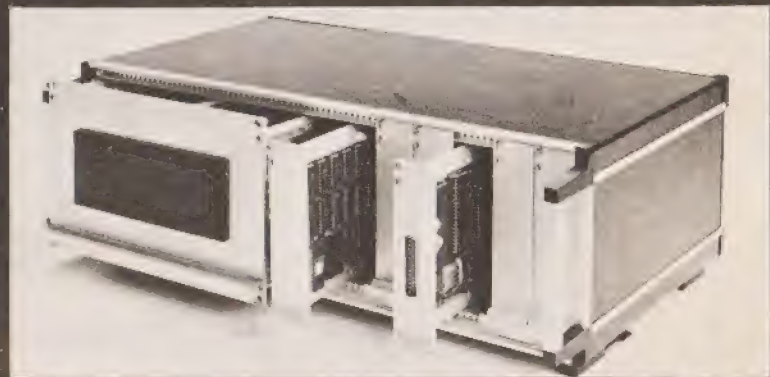


All Acorn modules are guaranteed and full after sales and technical advice services are available.



Acorn Memory

The first in our series of expansion cards is the Acorn 8K + 8K "state of the art" memory module. On a matching eurocard it provides 8K of ram (2114) and 8K of Eprom (2732) or 4K of Eprom (2716). It requires a single 5V rail, is designed for direct connection via a 32 way edge connector to the Acorn bus and is fully buffered for wiring into any system. Two onboard sockets provide independent positioning of Eprom and ram.



Software available soon includes 4K-Editor-Assembler-Disassembler, 4K Proprietary Fast Basic, Disc operating system with full file handling.
Although a standard strip of veroboard is all that is required for a full backplane, a racking system can be made available by Acorn Computers. The rack shown includes the VDU interface, two memory cards and dual floppy disc interface.

Order form

Send to: Acorn Computers Ltd. 4A Market Hill, Cambridge, Cambs.

☐ (qty) Acorn Microcomputer(s) in kit form at £65.00 plus £5.20 VAT
☐ (qty) Acorn Microcomputer(s) assembled and tested at £75.00 plus £6.00 VAT
☐ (qty) Acorn controller(s) (minimum configuration) at £35.00 plus £2.80 VAT
☐ (qty) Acorn Memory(s) assembled and tested at £95.00 plus £7.60 VAT
N.B. Price shown is for full 8K of ram, prices for smaller memory options and Eprom additions available on request.

I enclose a cheque for £............ made out to Acorn Computers Ltd.

Name ______________________
Address ______________________
____
____

Regd No 1403810

ACORN COMPUTER

Mr. P. Cornes

More details on last months NIM game

Last month we presented a program listing for the game of NIM; this month we will examine this program in some detail to see how it works. The first thing to do is to look at the winning strategy as this, after all, is the strategy that the computer should adopt. The winning strategy for the game of NIM is quite well known, and can be found in several maths textbooks on games theory (try Methematical puzzles and Diversions by MARTIN GARDENER). Due to the lack of space, this strategy will only be stated and not derived.

Winning Combinations

The first thing to do is to convert the number of matches in each pile into binary. As there are up to 7 matches in each pile, three binary digits are required for this conversion. The next thing to do is to add together all the first digits of the binary numbers produced, then add all the second digits together, and all the third digits. All these additions are done in decimal. When this is done, you are left with three decimal integer answers (see table 1). In our example, these are 4, 3 and 3. If any of these three digits is odd (which two of ours are) then the person next to play is in a winning position. The object now is for that player to remove some matches such that when this calculatiuon is done again, all the three numbers are even so that, for example, removing 3 from pile 1 would leave the three digits as 4, 2 and 2 (see table 2) which is thus a losing position for the player whose turn it is to play next.

All Is Revealed

Having looked briefly at the winning strategy which the computer adopts, let us now go on to look at the program presented as figure 1 last month. It would help if you could have that article in front of you as you read this description.

The program can very conveniently be broken into 5 main sections.

Section 1 is a subroutine which prints the current position of the board and also checks for a winning play. (Program lines 5000-5080. Flowchart Fig. 1).

PILE NO.	MATCHES IN PILE	BINARY		
		4	2	1
1	3	0	1	1
2	4	1	0	0
3	5	1	0	1
4	6	1	1	0
5	7	1	1	1
	COLUMN TOTALS	4	3	3

TABLE 1

PILE NO.	MATCHES IN PILE	BINARY		
		4	2	1
1	0	0	0	0
2	4	1	0	0
3	5	1	0	1
4	6	1	1	0
5	7	1	1	1
	COLUMN TOTALS	4	2	2

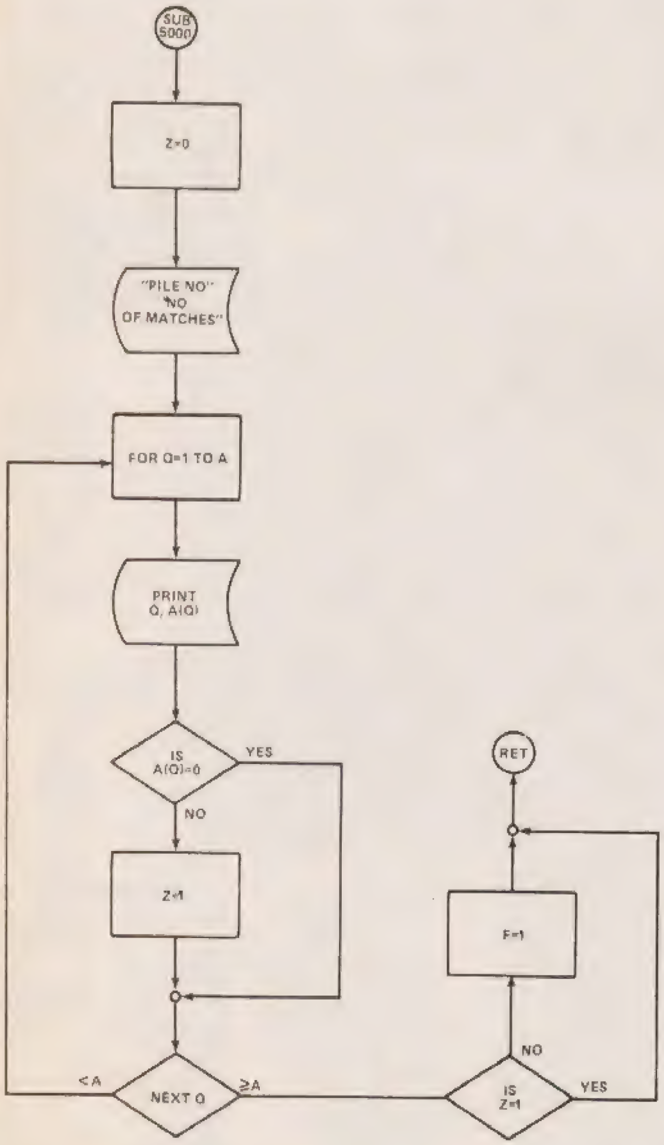
TABLE 2

Section 2 is also a subroutine and what this does is to convert the number of matches in each pile into binary and add them up as described earlier. It then makes a check to see of the computer is in a winning position or not. (Program lines 6000-7000. Flowchart Fig. 2).

Section 3 is where the program starts, and it initialises the values of variables and sets up the board. (Program lines 10-160. Flowchart Fig. 3).

Section 4 deals with the computers' opponents' move. (Program lines 163-180 and 2000-3030. Flowchart Fig. 4).

Fig. 1. Correct board position subroutine



Section 5 in conjunction with Section 2 enables the computer to evaluate and play its moves. (Program lines 190-360. Flowchart Fig. 5).
We will now go through each of these sections in turn in more detail.

Section 1

If you look at Flowchart Fig. 1, you will see that we print a heading (Pile No.   No. of matches) and then set up a FOR NEXT loop to print the pile numbers and the number of matches in the piles under this heading. The piles themselves are stored in A array locations A(1) to A(A) (where 3   A   6) and notice that after each pile has been printed, it is checked to see if it is empty. If it is, then we branch round to the NEXT Q statement. You should see from this that if all the piles are empty the variable Z will still have the value 0 that it received at the start of the routine. After all the piles have been printed, we test the value of Z and if it is 0 we set F to 1. This means that when we jump to this subroutine to print the board, a check is also made to see if the move just played has enabled one of the players to win the game. If it has, we set F to 1 to signify this fact (this technique is called setting a flag to show that an event has occurred).

Section 2

What this subroutine is doing (see Fig. 2) is to say, "if the computer were to take R matches from pile P, would that leave the opposition in a losing position." The first thing to do is to take R matches from pile P and this is done in the first box. We then set three variables to 0, (Z, U and I) which will be used later to keep a decimal total of the binary digits. The next box sets up a FOR NEXT loop in Q from 1 to A (A is the number of piles). The next step is to convert the number of matches in each pile in turn into binary (see last month's homework answer), the three binary digits being stored in variables V, B and M. We then add V, B and M to Z, U and I respectively to keep a running decimal total of the binary numbers. The next box we encounter is NEXT Q, which branches us back to deal with the provision of the next pile.

When all the piles have been converted to binary and added up into Z, U and I, we put R matches back into pile P and then check each of the digits Z, U and I in turn to see if it is odd or even. If any of these digits are odd, we branch to the RETURN statement. Only if they are all even will the variable O be set to 1 to flag the fact that taking R from pile P is a winning move for the computer.

Section 3

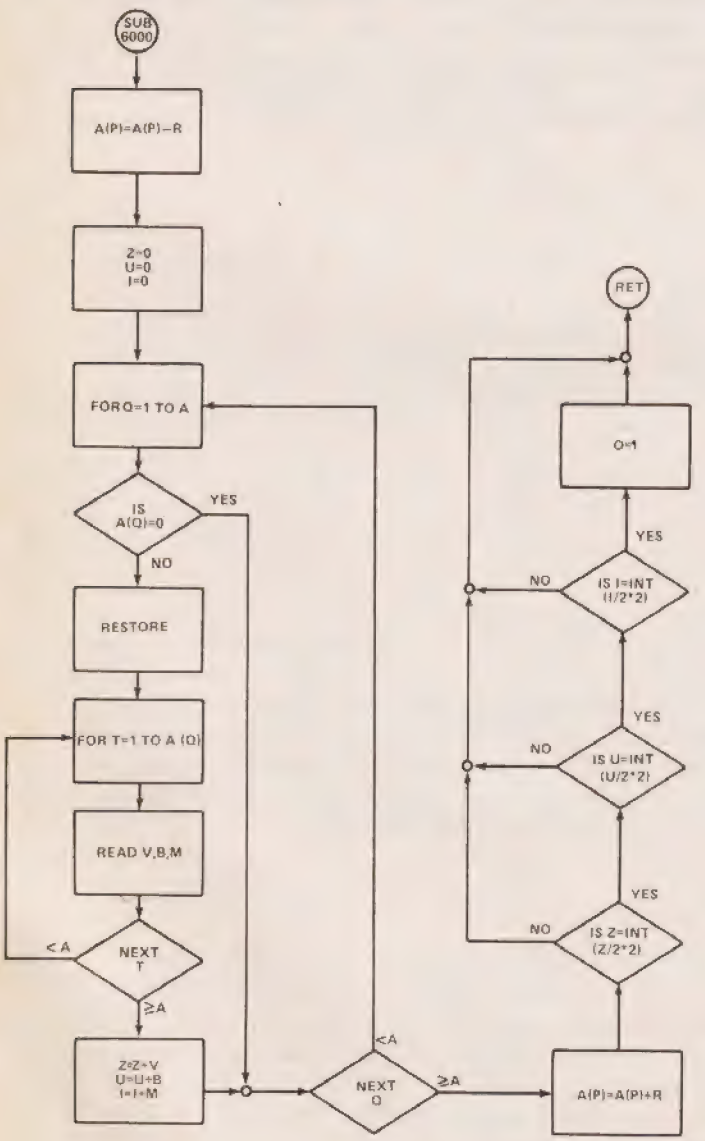
This is the start of the game proper (see Fig. 3) and the first question asks who should set up the board for the game.

If you wish to set up the board, you type 1; if the computer is to set it up, you type 0. We then come to the "IS A = 0" decision box. If the answer is YES, we branch right and the computer will pick a random number of piles (variable A such that 3 < A > 6) and put a random number of matches in each pile (program lines 150-160). If the answer to "IS A = 0" is NO, we continue down the flowchart and a question and answer session follows which allows you to set up the board (program lines 30-90). Which ever route we take, we end up at marker A and move on to flowchart Fig. 4.

Section 4

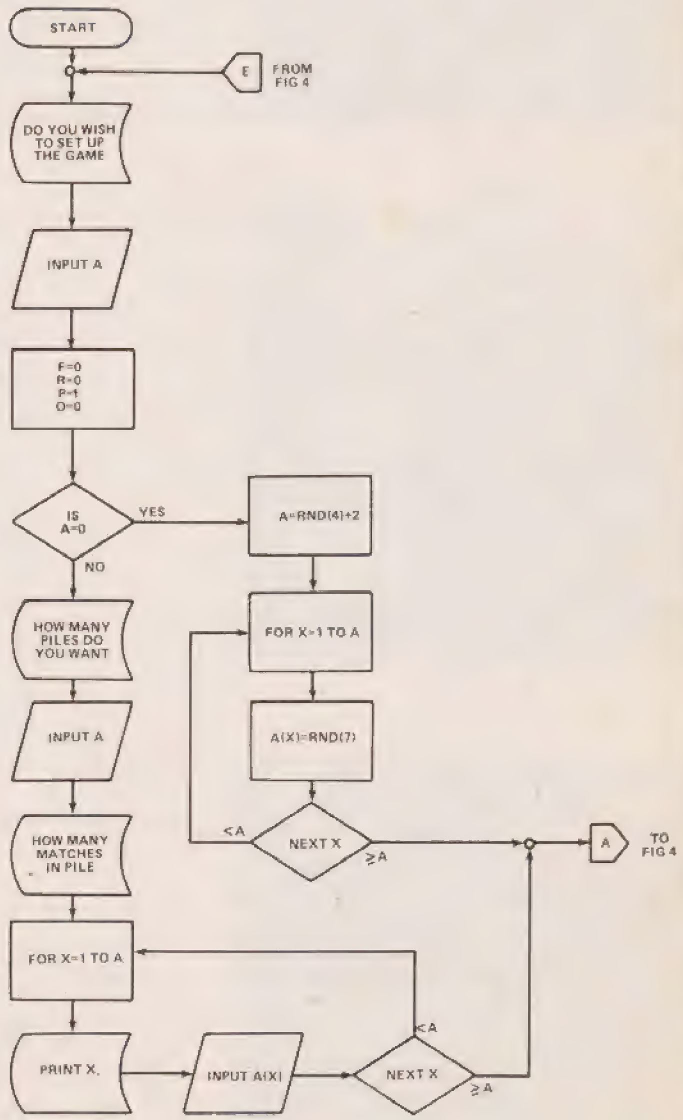
The first box in Fig. 4 jumps to subroutine at line 5000 (Fig. 1) and prints out the position of the board that has just been set up in section 3. It is not expected that F will be set to 1 at this point, and so it is not checked. The next three boxes deal with the question of who should start first, and Fig. 4 shows what happens if you choose to move first. You are asked which pile you wish to take from, and how many matches you wish to take from this pile. These matches are then removed (program lines 2000-2035). We then branch to subroutine at line 5000 (Fig. 1) to print the board and since a move has

Fig. 2. Pile check subroutine



DATA 0, 0,1,0,1,0,0,1,1,1,0,0,1,0,1,1,1,0,1,1,1

Fig. 3. Board set up routine



been made, the flag F is tested upon return. If the flag is set (which it shouldn't be after only one move) you are told you have won, and asked if you wish to play again. If you do, the program branches via marker E backk to the start box in Fig. 3. If not, the program ends. Assume, however, tht your first move did not enable you to win! (The reason for the check at this point is because this section is used every time you make a move and sooner or later you may well win). The program then branches to marker B in Fig. 5 which is the same place as you would have reached if you had decided to let the computer play first.

Section 5

This section works out the computer's move (see Fig. 5) and when you see the amount of work done, you will also see that it takes full advantage of the computer's speed. What happens in effect is that the computer starts with pile 1 (P = 1) or the first pile that contains matches (A (P) > 0) and takes 1 match from th pile (R = 1). A branch is then made to the subroutine at line 6000 (Fig. 2) where a check is made to see if this is a winning move (O = 1). If it is, then this move is made. If it is not a winning move, then another match is taken from the pile (R = 2) and the check is made again. Checking continues through all the matches (R = 1TO A[P]) in this pile and if no winning move is found, the computer moves on to the next pile that contains matches and checks all the possibilities in this second pile. This continues until either all the piles have been checked and no winning move is found, or until taking R matches from pile P produces a winning position (O = 1). If O ever equals 1, that move is made. If no winning move is found, then a move is made at random. After the computer has made its move, we GOSUB 5000 to print the current board and to check for a win. If the computer's move has given it the game (F = 1) this fact is printed, and we branch to marker D in Fig. 4 (Do you wish to play again?). If the computer has not won by its move, we branch to marker C in Fig. 4 for you to make your next move. At this point, I am taking my 23rd aspirin, and gratefully declare that to be it for this month. Probably the best homework for this month would be to go through and master the above program and flowcharts.

Next month, we'll take a break and make a start on extended BASIC.

Fig. 4. Opponents move routine

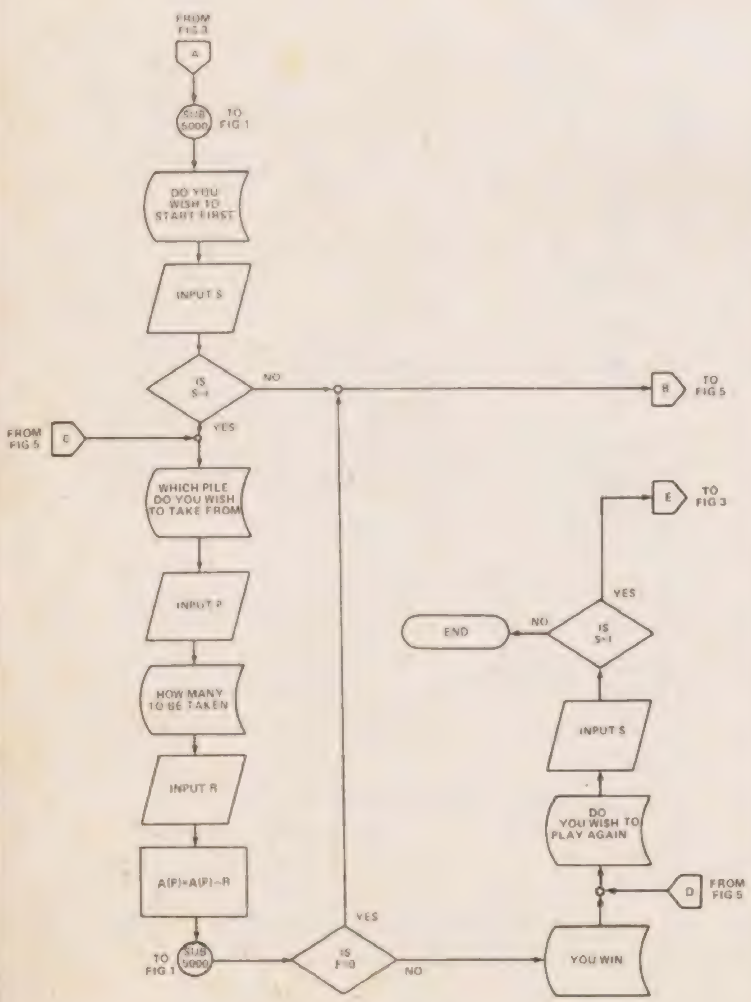
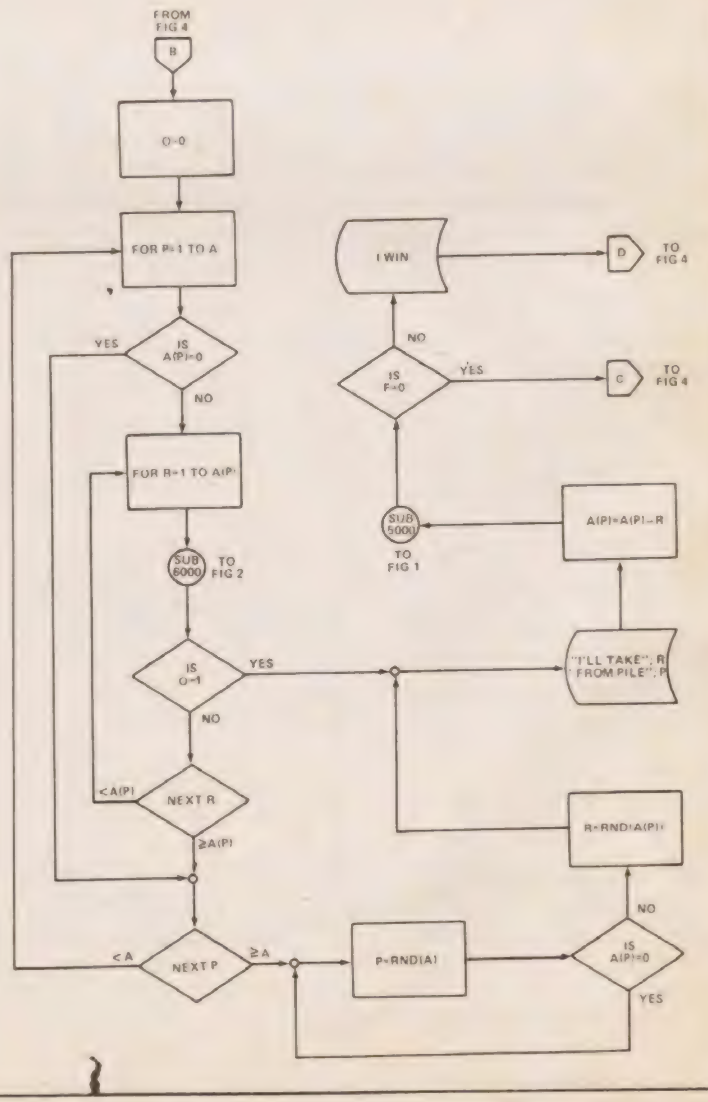


Fig. 5. Computer move evaluation routines



computing today

WHAT TO LOOK FOR IN THE JULY ISSUE ON SALE 15TH JUNE

MPUs By Expt.

One of the problems of becoming heavily involved in the software size of MPU technology is that the sheets of printout tend to obscure the hardware producing it.

Its a shame that more micro-men are not more knowledgable about the centre of their universe is it not? Naturally CT is doing something to redress this imbalance and next month we begin our "Microprocessors By Expt" series designed to lead one and all down the binary path.

Show report.

Spending a weekend in Florida may sound like a nice holiday but our faithful reporter brought home the goods on a micro-show. In this pictorial report we reveal the American way of showing off the goods.

Bits, Bytes and Pieces

Another look at the world of micro-electronics in our occasional series. The Vice President and General Manager of Motorola discusses the future of electronic components.

Superboard II

It was the first (commercially available) single board computer we heard about — ETIs Triton was really the first in Britain — but as yet it has not been considered seriously by the enthusiasts — or magazines (shame on us) — in this country and we thought it was about time that CT came riding over the hill and provided a full review to save the day! Next month we reveal all there is to know about this one PCB machine.

Details on one of the most misunderstood bus structures around

The pin designation and printed circuit card size originally used in the Altair 8800 computer has become somewhat of a standard in both the US personal computing scene and elsewhere in the world. So much so that there are over 40 manufacturers supplying electronic and mechanical products for the Altair bus. The bus has picked up a new name during this flurry of interest and is now being called the S100 bus.

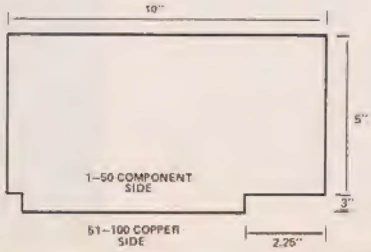
Originally defined by MITS when they designed the Altair 8800 computer around the 8080 microprocessor, the S100 gained momentum as a 'standard' when IMSAI released their computer, again 8080 based, with this bus structure. Now the range of boards available includes memory expansion, the most common floppy disc controllers, graphics boards, multifunction and analogue I/O boards, video interfaces and even speech synthesis and recognition systems. 1977 saw a new dimension added to the S100 bus. This was the year when various other CPU's became available as S100 structured boards. These included the Z80, the 6800 and the ever popular microprocessors 6502.

Expandability?

The huge variety of S100 format boards available has been largely responsible for its growing popularity. This wide choice has meant that you can tailor your own system by your needs by simply choosing the cards that you want. for example the addition of a graphics facility. You either spend a fortune on a graphics terminal or buy yourself a graphics board for your S100 system and not only save money but get a more flexible solution to your needs.

However the S100 system is not without its critics, not that other bus systems are perfect, and the real source of most of the arguements about the system are caused by the lack of information available on it. This lack of knowledge is not a very good thing when one is in the position of spending several tens of pounds on an add-on or hundreds on a system. At the present time the personal computer buffs are doing one of three things with the S100, ignoring it completely and using something else, buying a strict S100 system with proven boards and thirdly using the bus but with their own interpretation of the controls to suit their own microprocessor. Obviously the first and second groups will not have too much trouble but the third will need information on the bus structure. That information is what we hope to provide in this article.

Fig. 1. Standard S100 cards are 10" by a nominal 5.3". Some manufacturers change the height depending on circuit requirements. Edge connector spacing is 0.125", offset to prevent backward insertion of a board.



Criticisms

The bus is also criticised because it is not completely standardised, the control signals are getting to be rather outdated and more complex than they need to be. The justification for the claims stated above is that when the manufacturers look at the specification for the device that they are building some control lines can be interpreted in different ways and hence although variations do exist the boards work. What worries the users is that whilst board A may work on the bus and board B may work on the bus board A will not necessarily work with board B! The only way round this confusion is to try out the board before actually parting with your money and here your users club will probably be able to help.

As the S100 bus was designed around the 8080 when several of its support chips such as the 8228 and 8224 were not available there are more control signals on the bus than absolutely necessary. One good example is the provision of three lines to send the two clock phases and CLOCK. The newer processors have the oscillators built into the chip, others use only a single phase clock. However this surplus of control lines does have one advantage, that is the redundancy of information. To the system designer this is the flexibility that he can build into his system. The bus also has some unique control lines that he can build into his system. One example is the provision of a remote memory protect. The S100 defenition allows for a memory protect flip-flop on the memory board. Applying a momentary positive pulse on the MEMORY PROTECT line sets this flip-flop and prevents data being written into memory on that board.

Designing Your Own S100

The physical facts of S100 are given in Fig. 1, a picture worth a thousand words. The bus supports sixteen address lines, allowing 65536 bytes of memory to be uniquely addressed. There are also two 8-bit data buses, one for data input (data flowing to the CPU) and one for data output (data flowing from the CPU to memory or peripherals).

There is also a set of control lines that are used for synchronisation, timing data flow control and status control because the Altair is an 8080 based computer many of the control signals can be found described in the Intel data sheets. They can be found by looking at the similarity of names in Table 1 to those in the Intel User's Manual.

The S100 Definition calls for each board to have its own voltage regulators. To this end there are lines carrying unregulated voltages. There is +8V on pins 1 and 51, +16V on pin 2 and −16V on pin 52. Ground is pins 50 and 100.

When you examine the different signals in Table 1 you will notice frequent reference to the front panel. In fact many of the controls signals are generated on it. The Altair 8800 required the front panel to control the CPU board. However the newer CPU boards do not use front panel boards and themselves generate most of the control signals that are required by the S100 bus. This has happened through the use of an on-board ROM

monitor program. You will need to bear this in mind if you plan to design and build your own CPU card.

S100 Bus Structure

The S100 Bus Structure consists of 100 lines. These are arranged with 50 on each side of the plug-in cards.

The 'P' prefix indicates a processor command or control signal while the 'S' prefix indicates a processor status signal. All bus signals with the exception of the power supplies are TTL levels.

BUS DEFINITION

Table 1. The bus signals on an S100.

PIN No.	SYMBOL	NAME	EXPLANATION
1	+8V	+8 Volts	Unregulated input to +5 V regulators
2	+16 V	+16 Volts	Positve unregulated voltage
3	XRDY	External Ready	For special applications: pulling this line low will cause the processor to enter a WAIT state and allows the status of the normal Ready line (PRDY) to be examined.
4	V10	Vectored Interrupt Line 0	
5	V11	Vectored Interrupt Line 1	
6	V12	Vectored Interrupt Line 2	
7	V13	Vectored Interrupt Line 3	
8	V14	Vectored Interrupt Line 4	
9	V15	Vectored Interrupt Line 5	
10	V16	Vectored Interrupt Line 6	
11	V17	Vectored Interrupt Line 7	
18	STA DSB	Status Disable	This input to the CPU board tri-states the buffers that output the status information to the bus. Signals affected are SINTA, SWO, SSTACK, SHLTA, SOUT, SMI, SINP, and SMEMR.
19	C/C DSB	Command/Control Disable	This input to the CPU board tri-states the buffers that output the 8080 control signals to the bus. Signals affected are SYNC, DBIN, WAIT, WR, HLDA and INTE.
20	UNPROT	Unprotect	Is an input to the memory protect flip-flop on a memory board. To protect the contents of such boards a positive pulse should be applied to pin 70 to set the protect flip-flop. A positive pulse on the UNPROT line will reset the flip-flop.
21	SS	Single Step	This signal indicates the processor is performing a single tep. It comes from the front panel and is an input to the CPU.
22	ADD DSB	Address Disable	This input to the CPU tri-states all 16 address buffers and so isolates the 8080 adddress bus from the system address bus.
23	DO DSB	Data Out Disable	This CPU board input tri-states the data out buffers. Use of the signals on pins 18, 19, 22 and 23 effectively disconnect the CPU board from the system for DMA.
24	$\phi 2$	CLOCK PHASE 2	
25	$\phi 1$	CLOCK PHASE 1	
26	PHLDA	Hold Acknowledge	This CPU output indicates that the 8080 has entered the hold state and that the address and data outputs of the chip have gone tri-state (though not necessarily their buffers).
27	PWAIT	Wait	CPU output indicating 8080 in wait state.
28	PINTE	Interrupt Enable	CPU output indicating that the 8080 interrupt system is enabled and the chip will respond to interrupts.
29	A5	Address Line 5	
30	A4	Address Line 4	
31	A3	Address Line 3	
32	A15	Address Line 15	
33	A12	Address Line 12	
34	A9	Address Line 9	
35	DO1	Data Out Line 1	
36	DO0	Data Out Line 0	
37	A10	Address Line 10	
38	DO4	Data Out Line 4	
39	DO5	Data Out Line 5	
40	DO6	Data Out Line 6	
41	D12	Data In Line 2	
42	D13	Data In Line 3	
43	D17	Data In Line 7	

44	SM1	M1	CPU status output; indicates instruction fetch cycle (important for front panel operation as machine must halt on M1).
45	SOUT	OUT	Indicates execution of an OUT instruction: address bus contains I/O port address and data bus will contain output data when PWR active. All memory boards should be disabled when SOUT or SINP at logic 1.
46	SINP	INP	As SOUT, but for an IN instruction. Data to be input should be placed on the data bus when PDBIN is active.
47	SMEMR	MEMR	CPU output indicating memory read in progress.
48	SHLTA	HLTA	CPU status output: halt acknowledge.
49	CLOCK	Clock	In the Altair this is the inverted output of the 2 MHz oscillator that generates the two phase clock. However, other S100 cards are not staying with 2 MHz. The Morrow, for instance, outputs 18 MHz from its 8224 clock driver, while Z-80 and other systems differ again.
50	GND	Ground	You have now read half way through this; congratulations on your perseverance and I hope you find what you're looking for!
51	+8 V	+8 Volts	See Pin 1
52	—16 V	—16 Volts	Negative unregulated voltage.
53	SSW DSB	Sense Switch Disable	CPU input; disables data input buffers so that data from the front panel sense switches may be strobed onto the processor's bidirectional data bus.
54	EXT CLR	External Clear	Generated by the front panel; is used by the Altair as a reset signal for I/O devices. In other systems it is tied together with RESET and POC.
55-67			Are currently undefined on the Altair systems. Howver, a number of proposals have been put forward for their use. One proposal calls for a real time clock on pin 55 and the use of 56 — 60 as memory board selects. This would allow memory expansion in banks. Another proposal calls for 56 to be a strobe signal obtained from the 8224 clock chip and for pins 62 — 66 to be used for interface mass memory. For the time being, these pins are fair game for any special signals your system may require.
68	MWRT	Memory Write	A function of WR and SOUT, indicating data on data out bus to be written into memory.
69	PS	Protect Status	An output from the memory board currently being addressed, indicates status of memory protect flip-flop.
70	PROT	Protect	Is the input to the memory protect flip-flop on the board currently addressed.
71	RUN	Run	Indicates the state of the RUN/STOP flip-flop.
72	PRDY	Ready	CPU board input that controls the run state of the processor. Pulling PRDY low causes the processor to enter a wait state until PRDY goes high again.
73	PINT	Interrupt Request	
74	PHOLD	Hold	Causes the processor to enter a Hold state and subsequently acknowledge by putting PHLDA high.
75	PRESET	Reset	Resets program counter to zero.
76	PSYNC	Sync	Identifies beginning of a machine cycle.
77	PWR	Write	Indicates dat is being written to memory or I/O. Data on bus is stable while PWR is low.
78	PDBIN	Data Bus In	Processor output control signal indicating that data is being read into the CPU. Data on the data bus should stable while PDBIN is high.
79	A0	Address Line 0	
80	A1	Address Line 1	
81	A2	Address Line 2	
82	A6	Address Line 6	
83	A7	Address Line 7	
84	A8	Address Line 8	
85	A13	Address Line 13	
86	A14	Address Line 14	
87	A11	Address Line 11	
88	DO2	Data Out Line 2	
89	DO3	Data Out Line 3	
90	DO7	Data Out Line 7	
91	DI4	Data In Line 4	
92	DI5	Data In Line 5	
93	DI6	Data In Line 6	
94	DI1	Data In Line 1	
95	DI0	Data In Line 0	
96	SINTA	INTA	Indicates interrupt acknowledge.
97	SWO	WO	Processor output indicating write cycle.
98	SSTACK	Stack	Processor output indicating that the address bus holds the stack pointer.
99	POC	Power On Clear	When mains is first applied this signal is generated to set up initial conditions on other boards in the system.
100	GND	Ground	

Mr. M. Bell

An educational package for your NASCOM in our continuing series

For any personal computer owner, one of the fundamental requiremets of the system he owns is its justification on cost grounds. Few of us can afford £200+ for any item that will remain just a 'toy'. This is never more true when the proud owner is a family man with so many other important expenditures. To date there are very few programs written for constructive uses of the computer.

I have attempted to create, within the limited memory capacity of a "bog" standard NASCOM computer, a general purpose educational program that can be used for a multiplicity of subjects for all age groups. The parameters of the program were to be as follows:

1) The main program should be as flexible as possible.
2) It should occupy as little memory space as could be achieved to leave the maximum room for questions.
3) Where possible the questions should be reversible, ie. What is the Italian word for? Tradorre in Inglese?
4) The questions should be selected at random and only used once.
5) The answers should be 'echoed' onto the VDU
6) Backspace correction is essential
7) The completed answer to be indicated by typing N/L.
8) A wrong answer of a pass should be corrected immediately.
9) A score of 'Correct', 'Wrong' and 'Pass' should be displayed on completion or termination of the exercise.

Realisation of all these requirements was achieved with room for approximately 30 questions. This is thought to be sufficient to make the program viable.

The version is for teaching Italian, although the text has been written to teach under 5's to count and adults language vocabulary. Other texts included are English grammar examples.

Initialisation of the program clears the screen of unwanted characters, sets the score registers to zero and scans through the library setting the sign character to 28H. Selection is then carried out using the 'pseudo random number' instruction on the Z80, machine code ED 5F. Whilst not truly random it is adequate for our needs. This number is added to the previous library address to give a new address. It is unlikely that the new address will coincide with the start of a 'pair' so a search is carried out to find the first free one after that location. Should the end of the library be reached the 'end character' is received (25H) and the search continues from the library start address. The LSB of the randon number is also used to determine which way round the question should be asked. (see requirement 3).

The question is now printed on the VDU and the answer is awaited. Each typed character is echoed onto the VDU and can be corrected using the B/S key. On receipt of the N/L character the answer is checked against the relevant library entry. If the answer is wrong, or if the pass character was typed the correct answer is printed beneath the incorrect one. A correct answer will result in the next question being asked.

When the exercise is complete the score is displayed at the bottom of the screen and the processor is HALTED. To do again the RS key should be operated and execution commenced from address OC50.

A number of modifications can be made to this program to suit local requirements.

1) If a relay is fitted to START/STOP a tape dump then whilst the score is being displayed further questions can be automatically loaded.
2) If the NASCOM is expanded the number of questions available is limited only by the amount of RAM available and the size of the score registers.

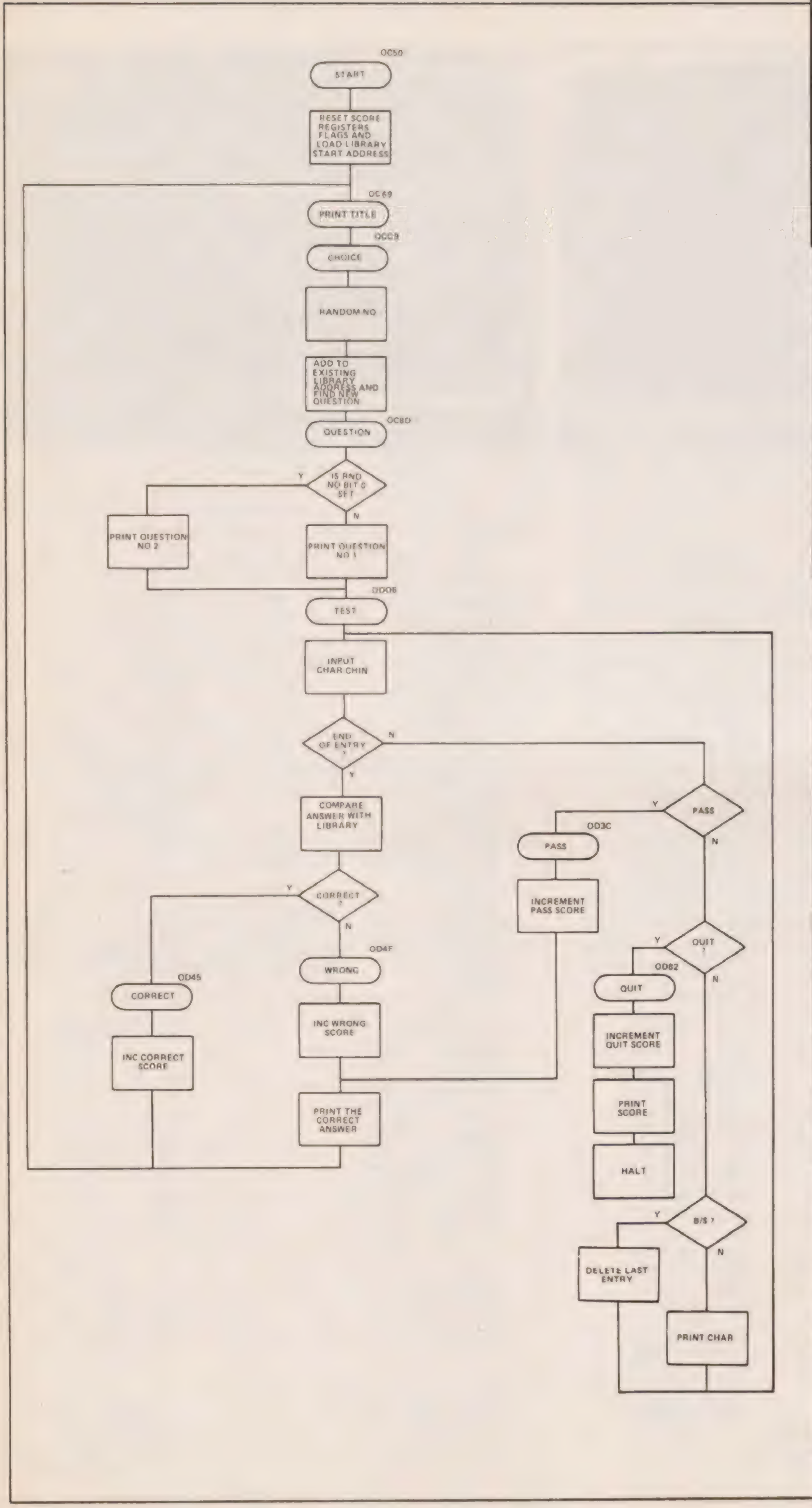
As usual the competence of personal computer users is bound to bring forth modifications and additional subject matter, and I will be pleased to hear of any developments in this direction.

```
C50 21 03 0D     'START'    HL = 0D03
C53 AF                      XOR A
C54 77                      (HL) ,A
C55 23                      INC HL
C56 77                      (HL) , A
C57 23                      INC HL
C58 77                      (HL) , A
C59 21 29 0E                HL = 0E29
C5C E5                      PUSH HL
C5D 00                      NOP
C5E 23           'INIT 1'   INC HL
C5F 7E                      A , (HL)
C60 FE 29                   CP = 29
C62 20 01                   JRNZ – 'INIT 1'
C64 35                      DEC (HL)
C65 FE 15        'INIT 2'   CP = 15
C62 20 01                   JRNZ – 'INIT 2'
PRINT TITLE
C69 3E 1E                   A = 1E
C6B CD 3B 01                Call CRT
C6E 01 DA 0B                BC = 0BDA
C71 21 C6 0D                HL = 0DC6
C7C CD B8 0D                Call 'PRINT'
```

```
C77 18 50                   JR – 'CHOICE'
SUB 1
C79 01 4C 09                BC = 0950
C7C CD B8 *d                Call 'PRINT'
C7F C5                      PUSH BC
C80 D9                      EXX
C81 C1                      POP BC
C82 CD B8 0D                Call 'PRINT'
C85 C5                      PUSH BC
C86 D9                      EXX
C87 C1                      POP BC
C88 23                      INC HL
C89 CD B8 0D                Call 'PRINT'
C8C C9                      RTN
QUESTION
C8D CB 41                   Test C bit O
C8F 20 0A                   JRNZ 'CAP'
C91 EB           'COUN'     EX HL/DE
C92 D9                      EXX
C93 21 ED 0D                HL = ODED
C96 CD 79 0C                Call 'SUB 1'
C99 18 07                   JR – 'INS'
C9B D9           'CAP'      EXX
```

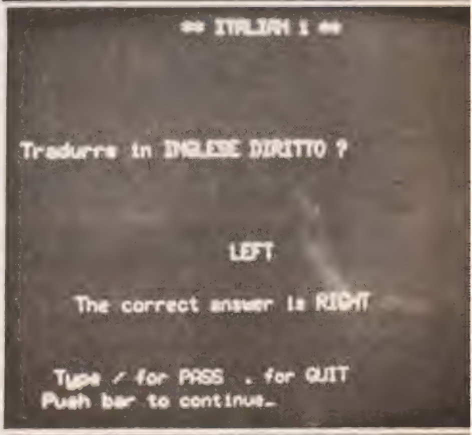

```
C9C 21 0E 0E                 HL = 0E08
C9F CD 79 0C                 Call 'SUB 1'
CA2 EF                       STRING
CA3  Type / for PASS., for QUIT □
CC3 CD 40 02                 Call CRLF
CC6 C3 06 0D                 JP — 'TEST'

CHOICE
CC9 16 25                    D = 25
CCB E1                       POP HL
CCC ED 5F                    RANDOM No
CCE 47                       B , A
CCF 4F                       C , A
CD0 CB 89                    RST C Bit 1
CD2 7A         'FIND 1'      A , D
CD3 23         'FIND 2'      INC HL
CD4 BE                       CP (HL)
CD5 20 05                    JRNZ 'FIND 3'
CD7 21 2A 0E                 HL = 0E2A
CDA 18 F6                    JR — 'FIND 1'
CDC 10 F5      'FIND 3'      DJNZ — 'FIND 2'
CDE 23         'FIND 4'      INC HL
CDF 3E 28                    A = 28
CE1 BE                       CP (HL)
CE2 28 10                    LRZ — 'FIND 5'
CE4 7A                       A ,D
CE5 BE                       CP (HL)
CE6 20 F6                    JRNZ — 'FIND 4'
CE8 CB 49                    TEST C Bit 1
CEA C2 82 0D                 JZ - 'QUIT'
CED CB C9                    Set C Bit 1
CEF 21 C5 0D                 HL = 0DC5
CF2 18 EA                    JR — 'FIND 4'
CF4 CB 89      'FIND 5'      RST C Bit 1
CF6 34                       INC (HL)
CF7 23                       INC HL
CF8 E5                       PUSH HL
CF9 D1                       POP DE
CFA AF                       XOR A
CFB 23         'FIND 6'      INC HL
CFC BE                       CP (HL)
CFD 20 FC                    JRNZ — 'FIND 6'
CFF 23                       INC HL
D00 E5                       PUSH HL
DO1 18 8A                    JR — 'QUEST'
D03 1 register for correct answer score
D04 1 register for wrong answer score
D05 1 register for pass score
```

```
TEST
DO6 D9                       EXX
DO7 EB                       EX HL , DE
DO8 E5                       PUSH HL
DO9 01 1E 0A                 BC = 0A1E
DOC C5                       PUSH BC
DOD CD 3E 00   'CHIN'        Call CHIN
D10 FE 1F                    CP = 1F
D12 28 16                    JRZ — 'N/L'
D14 FE 2F                    CP = 2F
D16 28 24                    JRZ — 'PASS'
D18 FE 2E                    CP = 2E
D1A 28 66                    JRZ — 'QUIT'
D1C FE 1D                    CP = 1D
D1E 20 06                    JRNZ — 'N/L 0'
D20 0B                       BEC BC
D21 3E 20                    A = 20
D23 02                       (BC) , A
D24 18 E7                    JR — 'CHIN'
D26 02         'N/L 0'       (BC), A
D27 03                       INC BC
D28 18 E3                    JR — 'CHIN'
D2A C1         'N/L'         POP BC
D2B 0A         'N/L 1'       A , (BC)
D2C BE                       CP (HL)
D2D 20 04                    JRNZ — 'N/L 2'
D2F 03                       INC BC
D30 23                       INC HL
D31 18 F8                    JR — 'N/L 1'
D33 AF                       A = O
D34 BE                       CP (HL)
D35 28 0E                    JRZ — CORRECT
D37 18 16                    JR — WRONG
D39 00                       NOP
D3A 00                       NOP
D3B 00                       NOP

PASS
D3C 21 05 D                  HL = OD05
DF3 CD C1 D                  CALL DAA
D42 E1                       POP HL
D43 18 10                    JR — ANS

CORRECT
D45 21 03 0D                 HL = 0D03
D48 CD C1 0D                 CALL DAA
D4B E1                       POP HL
D4C C3 69 0C                 JP — 'PRINT TITLE'
```



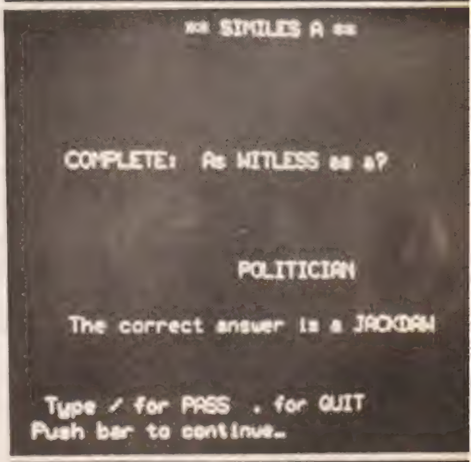
0C50
START
RESET SCORE REGISTERS FLAGS AND LOAD LIBRARY START ADDRESS
0C69
PRINT TITLE
0CC9
CHOICE
RANDOM NO
ADD TO EXISTING LIBRARY ADDRESS AND FIND NEW QUESTION
0C8D
QUESTION
IS RND NO BIT 0 SET
Y
N
PRINT QUESTION NO 2
PRINT QUESTION NO 1
0D06
TEST
INPUT CHAR CHIN
END OF ENTRY ?
N
Y
COMPARE ANSWER WITH LIBRARY
CORRECT ?
Y
N
0D45
CORRECT
INC CORRECT SCORE
0D4F
WRONG
INC WRONG SCORE
PRINT THE CORRECT ANSWER
PASS
Y
N
0D3C
PASS
INCREMENT PASS SCORE
QUIT ?
Y
N
0D82
QUIT
INCREMENT QUIT SCORE
PRINT SCORE
HALT
B/S ?
Y
N
DELETE LAST ENTRY
PRINT CHAR

** SIMILES A **
COMPLETE: As GRAVE as a?
Type / for PASS . for QUIT
SCORE RIGHT 10 WRONG 03 PASS 02.



** ITALIAN 1 **
Tradurre in INGLESE DIRITTO ?
LEFT
The correct answer is RIGHT
Type / for PASS . for QUIT
Push bar to continue.

** SIMILES A **
COMPLETE: As LOYAL as a?
BAT
The correct answer is a DOVE
Type / for PASS . for QUIT
Push bar to continue.



** SIMILES A **
COMPLETE: As WITLESS as a?
POLITICIAN
The correct answer is a JACKDAW
Type / for PASS . for QUIT
Push bar to continue.

** SIMILES A **
COMPLETE: As PRETTY as a?
PICTURE
Type / for PASS . for QUIT

```
WRONG
D4F AF                              XOR A
D50 21 04 0D                        HL = 0D04
D53 CD C1 0D                        CALL DAA
D56 21 D6 0D        'ANS'           HL = ODD6
D59 01 90 0A                        BC = 0A90
D5C CD B8 0D                        CALL PRINT
D5F E1                              POP HL
D60 CD B8 0D                        Call PRINT
D63 EF
D64     Push bar to continue □
D7C CD 3E 00                        Call CHIN
D7F C3 69 0C                        JP — PRINT TITLE

QUIT
D82 EF
D83     Score RIGHT  □
D93 21 03 0D                        HL = 0D03
D96 7E                              A , (HL)
D97 CD 44 02                        CALL B2HEX
D94 EF
D9B     WRONG  □
DA4 23                              INC HL
DA5 7E                              A , (HL)
DA6 CD 44 02                        CALL B2HEX
DA9 EF
DAA     PASS  □
DB3 23                              INC HL
DB3 7E                              A , (HL)
DB4 CD 44 02                        CALL B2HEX
DB7 76                              HALT

PRINT
DB8 7E                              A , (HL)
DB9 FE 00                           CP = O
DBB C8                              RZ
DBC 02                              (BC) , A
DBD 03                              INC BC
DBE 23                              INC HL
DBF 18 F7                           JR—PRINT
DAA
DC1 7E                              A , (HL)
DC2 3C                              INC A
DC3 27                              DA
DC4 77                              (HL) , A
DC5 C9                              RTN
```

ENGLISH 7

ANTONYMS A

```
DC6   * ANTONYMS A * □□
DD6   The correct answer is □
DED   What is the antonym of □
E05     ? □
```

```
LIBRARY
E08 (COLD □ HOT □              (ASLEEP □ AWAKE □
    (FIND □ LOSE □             (BUSY □ AWAKE □
    (BRIGHT □ DULL □           (TRUE □ FALSE □
    (UGLY □ PRETTY □           (POOR □ RICH □
    (COARSE □ FINE □           (ROUGH □ SMOOTH
    (PROUD □ HUMBLE □          (WIDE □ NARROW □
    (CRY □ LAUGH □             (BLESS □ CURSE □
    (ENTRANCE □ EXIT □         (ACTIVE □ PASSIVE □
    (LEND □ BORROW □           (JUNIOR □ SENIOR □
    (LIKE □ DISLIKE □          (SUCCESS □ FAILURE □
    (HARMONY □ DISCORD □       (ADVANCE □ RETREAT □
    (ARRIVE □ DEPART □         (ATTACK □ DEFEND □
    (ENGAGE □ DISMISS □        (APPEAR □ DISAPPEAR □
    (COLLECT □ DISPERSE □      (YOUTH □ AGE □
    (MODERN □ ANCIENT □        (FORMER □ LATTER □
    (SLENDER □ STOUT □         (OFTEN □ SELDOM □
    (HERE □ THERE □            (DAY □ NIGHT □
    %
```

ITALIAN 1

```
DC6   ** ITALIAN 1 ** □
DD6   The correct answer is □
DED   What is the Italian word for □
EOB     ? □
EOE   Tradurre in inglese □
E24     ? □□□□□
```

```
Library
E2A (MOTHER □ MADRE □          (FATHER □ PADRE □
    (SISTER □ SORELLA □        (BROTHER □ FRATELLO □
    (BOY □ RAGAZZO □           (GIRL □ RAGAZZA □
    (FRIEND □ AMIGO □          (WORK □ LAVORARE □
    (RED □ ROSSO □             (BLUE □ TURCHINO □
    (GREEN □ VERDE □           (YELLOW □ GIALLO □
    (BLACK □ NEGRO □           (WHITE □ BIANCO □
    (HOT □ CALDO □             (COLD     □     FREDDO     □
    (DAY □ GIORNO □            (NIGHT □ NOTTE □
    (SEA □ MARE □              (COUNTRY □ CAMPAGNA □
    (OFFICE □ UFFICIO □        (STATION □ STAZIONE □
    (LEFR □ SINISTRA □         (RIGHT □ DIRITTO □
    (HELP □ AIUTO □            (BED □ LETTO □
    (SAND □ SABBIA □           (YOU □ TU □
    (ME □ ME □                 (I □ IO □
    (THEY □ ESSI □             (PLEASE □ PIACERE □
    (THANK YOU □ GRAZIA □
    %
```

ENGLISH 4

GRAMMAR A

```
DC6   ** GRAMMAR A ** □
DD6   The correct answer is □
DED   What is the verb of the adjective □□□
E12     ? □
E15   What is the adjective of the verb     □□□
E3A     ? □
```

```
Library
E3D (STRONG□ STRENGTHEN□       (VACANT □ VACATE □
    (NEW □ RENEW □             (ELECTRIC □ ELECTRIFY □
    (SIMPLE □ SIMPLIFY □       (FALSE □ FALSIFY □
    (ANGRY □ ANGER □           (PURE □ PURIFY □
    (HUMBLE □ HUMILIATE        (SOLEMN □ SOLEMNISE □
    (STUPID □ STUPIFY □        (EQUAL □ EQUALIZED □
    (LARGE □ ENLARGE □         (POOR □ IMPROVERISH □
    (CALM □ BECALM □           (RICH □ ENRICH □
    (LIVELY □ ENLIVEN □        (FEEBLE □ ENFEEBLE □
    (SAD □ SADDEN □            (GLAD □ GLADDEN □
    (HOT □ HEAT □              (SAFE □ SAVE □
    (LONG □ LENGTHEN □         (NOBLE □ ENNOBLE □
    (CHEAP □ CHEAPEN □

    (CHEAP □NCHEAPEN □         (MAD □ MADDEN □
    (SMOOTH □ SMOOTH □         (MODERN □ MODERNIZE □
    %
```

EDUCATIONAL PACKAGE: ADDRESS VARIATIONS

	C5A	C8F	C90	C94	C9D	C9E	CD8
SMILES	06	00	00	F0	08	0E	07
GRAMMAR	3C	20	0A	ED	15	0E	3D
ANTONYMS	07	20	0A	ED	ED	0D	88
GERMAN	2E	20	0A	ED	0D	0E	2F
FRENCH	2E	20	0A	ED	0D	0E	2F
ITALIAN	29	20	0A	ED	0F	0E	2A
GEOGRAPHY	29	20	0A	ED	08	0E	2A

Note at the bottom of the Italian questions the change should be to 9D not 9A as shown.

BITS, BYTES & BAUDS

This month we take a look at mass storage memory systems

At this point in our series, we have seen how a computer works, how it addresses memory, and how it can send data to and receivee data from a peripheral. The program (or set of instructions for the computer to follow) must reside in the memory that is readily available to the computer so that it has access to all it needs, simply by sending an address on the address lines and receiving instructions on the data lines. If the computer in our example were to be doing a constant, repetitive, function such as controlling the traffic lights at an intersection, then it would be permanently burned into a ROM (Read-only-memory). This is the simpliest form of a computer system, and its description ends here.

Of more interest to us is the general purpose counter, such as may be found in an office, for example. This type of computer may have more than a hundred different programs for various day-to-day jobs, as well as dozens more that are executed only once a week, once a month, or less often. These programs are not all needed at the same time, and so need not all be stored in the main memory. Main memory is very expensive when compared to office memory, such as paper tape, cassettes, and discs, and so shrewd system designers store as many programs offline as possible to reduce main memory needs.

Online-Offline Storage

To get better ideas of the difference between outline and offline storage, let's compare our computer to a theatre. The actors who are currently performing occupy the stage, which is a very elaborate platform, designed to be in full view of the audience, and having lots of expensive accessories, such as lights, microphones, scenery etc, to make it a very effective place for the actors to work. When the actors are not acting, they leave the stage and wait in the wings. The wings on the stage are very crowded places where space is at a premium, and so you will only find them there immediately before and after their performance on the stage. Beyond the wings you will find all sorts of space reserved for dressing rooms, etc, where actors can go to relax. They know that when they are in these rooms, they will get plenty of notice before they are needed on the stage.

In a computer, the stage is the main memory, which is sometimes referred to as executable memory, because programs must be in this type of memory to be executed. It is very expensive, like the stage, because it is very fast, and programs are only put here when they are needed. The wings of the stage represent any form of fast-access device, such as floppy disk, where data can be stored and retrieved very quickly. Data and sub-programs that are used constantly by a main program in memory are stored on such a device, because during the course of a program's execution they may be brought into memory and taken out again as required by the program.

The dressing rooms of our computer are devices such as a cassette system where we can store programs until their turn comes to go into the main memory. Changing of main programs happens relatively infrequently, and lots of Notice can be given, so the delays in getting programs into memory aren't very serious.

Now that we have decided that it's smart to store programs and data offline when not using them, let's look into the multitude of devices that can be used for storage.

Old Faithful — The Paper Tape

For the small system owner just building his system, an obvious starting point is a teletype machine (TTY). TTYs sometimes have a punch and reader added, capable of storing and retrieving data by punching them on paper tape. For computer purposes, an 8 level machine is needed, such as the Teletype Corporation model ASR33 or ASR35. (ASR stands for Automatic Send-Receive, meaning that it is a basic unit with a reader-punch added, as opposed to a KSR which simply means Keyboard Send-Receive — i.e. no punch/reader).

In order to store all programs in a manner that can be readily reloaded into the computer, a standard must be established to define what each item on the tape means. Each part of the memory, as it is dumped on to punched tape, can be considered as a block. These blocks of memory be any convenient length, and are stored as individual records on the tape. Every byte of data, when sent to the teletype machine, will cause the eight hole-punches to pierce the paper or not, depending on whether the corresponding bit is 1 (hole) or a 0 (no hole). The paper moves forward by one byte, or frame has been punched, and at th same time the eletype machine is receiving the next character for punching. In this way, the result of sending a complete block memory, one byte at a time, will be a long tape with holes corresponding to the bits stored in memory as ones.

Holes Are Not Enough

However, just to store the contents of memory on to paper tape does not constitute a complete record. Generally-accepted practice dictates that prior to starting a dump of some memory, the computer shall send the following to the teletype: (Fig. 1).

a) A leader consisting of several frames (usually ten or more) of blank tape. This gives the user space to write on the tape describing the program, and alsao provides an alert to the reader (when the program is loaded into the computer again) that a new block follows.

b) One or two frames with a code that indicates that a new record is about to start, and sometimes, optionally, the type of record (data or program).

c) Two frames which specify the address where this program starts in memory.

d) Two frames that specify the address to which control should be given at the completion of loading if this is a self-starting program.

e) One or two frames which specify the length of this block of data, so that the loader knows when to expect the end of the data.

f) A variable number of frames containing the actual data taken from memory, beginning at the address specified in part (C).

g) A special frame or frames that contain a checksum,

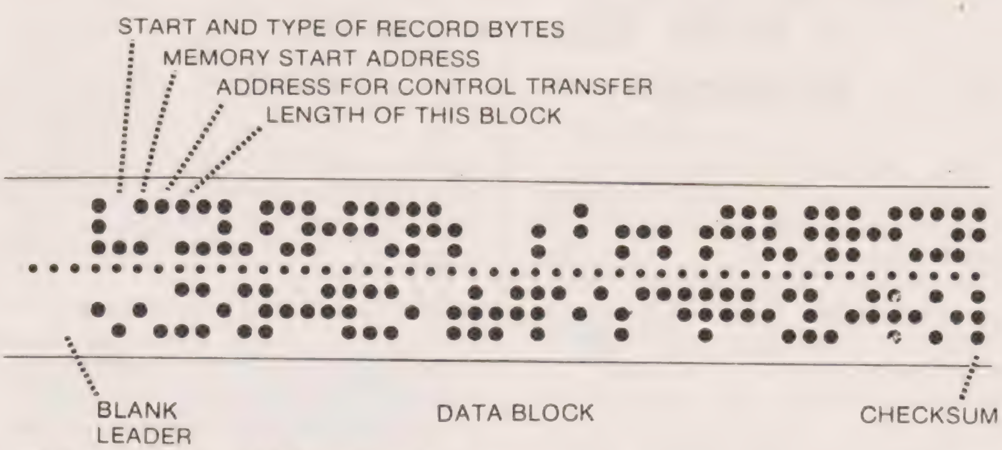


Fig. 1. Data and preainble/postainble layout on paper tape

compiled from a process which is affected by every byte of data on the tape. This byte is calculated and stored as the computer is dumping the program. When the computer reads in the program again, it makes the same calculation as it made before it dumped the data, except that this time it is basing the calculation on the data received from the tape. After the computer has made this calculation on all information read in, it then compares the answer it now gets to the answer that is stored as the last frame(s) on the tape. If the two answers disagree, then the opertor is warned that an error occurred during the reading back of the last block, and that he should retry that block.

h) Several rubouts, or all-holes-punched frames to indicate an end-of-block.

A long program would be broken dowm into many smaller blocks, so that if an error occurs during a single block, the block itself would be retried by the operator until it is received correctly, instead of having to wait until the entire program has read in before reloading it all.

The above program tape decription does not necessarily represent an actual format used by any particular manufacturer but rather a general idea of how most of them operate.

This method is similar to the main form of program loading in the early days of small minicomputer and microcomputer systems. It is largely obsolete now in the minicomputer field, except for small dedicated systems such as industrial controllers whose program is changed very rarely. It is still used, however, as a last-ditch method of getting dignostic programs into a computer system when a field engineer cannot get a major system device, such as a disc, to load programs. For this reason, most microcomputer manufacturers still ship diagnostics on paper tape with their products.

But Everybody Has A Cassette

A natural development from paper tape was the cassette, although this method has failed to fulfil the expectations of commercial users, since cassette transports designed for home entertainment use are not accurate enough for high-speed data use, and the development of purely digital systems was largely overshadowed by IBM's introduction of the floppy disc. A cassette can be viewed in exactly the same manner as paper tape, including the way that blocks are formatted as described above. The only difference is that instead of parallel holes being used to represent the ones and zeros, toes are used in pure serial fashion on the cassette, i.e. one after the other. The simplest method of recording data on a cassette would be to send the data to an asynchronous line, just as if it were going to be punched on a teletype machine, but replacing the teletype machine with a mjoden and recording the tones from the modem on tape. This method is rarely used, more modern and efficient methods such as CUTS and Kansas City standards having been developed by computer hobbyists for better reliability.

One major disadvantage of both the paper tape and cassette systems is that they are both sequential access systems. This means that if you store fifty programs, and later want to retrieve the fiftieth one, you have to read through the preceeding forty-nine programs until you get the one you want. With paper tape, this headache is almost eliminated by putting every program on a separate piece of paper tape, but this has the ·advantage of requiring operator intervention to find the correct tape and load it into the eader. Cassette systems can sometimes have a better approach to this problem. This type of system can turn off the tones completely for a distance between the data blocks, so that there is a quiet spot on the tape. The program puts a program name or identifier at the beginning of every block. (see Fig 2) When the cassette controller is instructed to find a particular block number or program name, it reads the first few frames of a block, checks to see if the identifier matches that which it is looking for, and if so, reads the whole block. If it does not match, it then fast-forwards the tape until another silent gap is detected, at which time the controller slows down the tape to read the first few frames, and so on until the whole program is loaded.

Enter The Floppy Disc

When IBM introduced the floppy disc, it was designed to be a quick, reliable method of reloading the microprogram of their big mainframes, quickly and effectively after a power failure or other cause of loss of control memory. As a result of the availability of the floppy disc it is not worth the trouble and expense for anyone to develop a general-

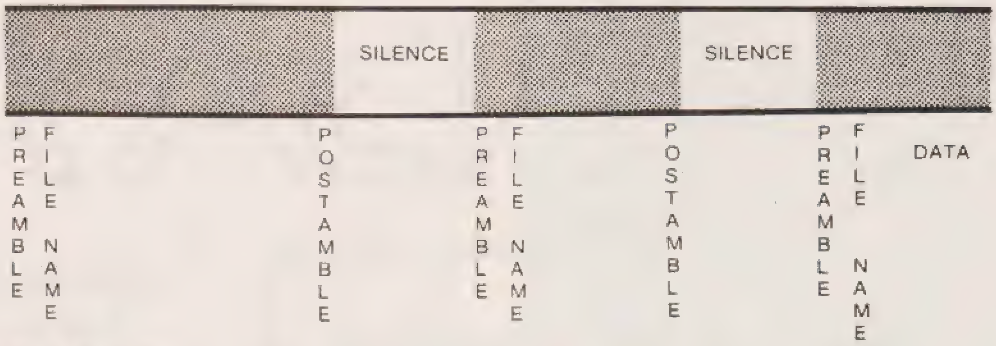


Fig. 2. Data files on a cassette

purpose, commercial-grade cassette system, since cassettes are inherently unsuited for data purposes and a floppy disc system is only marginally more expensive. Also, the floppy disc wins hands down over all sequential-access devices, because it is a random-access device.

Random access simply means that any part of a file can be accessed without having to read all data before it.

The floppy disc is made up of a flexible, circular piece of mylar, about the size of a 45 rpm gramophone record. (see Fig. 3) One side of the "diskette" is covered with metal oxide, just as is one side of a cassette tape. On this surface, all data are recorded, in the following manner:

As the diskette spins around at 369 rpm, the head can move in and out along a line joining the centre of the hub of the diskette and a fixed point on the perimeter of the diskette. The actual diskette is enclosed in an envelope to protect it, and a slot is cut in the envelope alone the line of travel of the magnetic recording head, on the side of the diskette containing the oxide. The inside of this envelope contains a felt-like material that cleans the diskette and traps any foreign particles as the diskette spins.

The head can be moved in and out over the surface of the diskette in increments of 1/48 inch. Every 1/48 inch is defined as being a track, starting at a predetermined distance in from the edge of the diskette. Each track forms a circle, centred on the centre of the disc. Thus all data are written on the 77 concentric tracks.

At a specified spot on the mylar, a hole is punched so that light can ıpass through it. This point is defined as being the start point of all the tracks. As the disc spins past this point, no matter which track the head is over, it will be at the beginning of that track.

Each track is further divided into 26 sectors. In some diskettes, a hole is punched in the mylar to signal the beginning of each sector. These are known as hard-sectored diskettes, since the hardware (photocell) actually gives a signal to the interface to tell it when a new sector is starting. The IBM 3740 standard diskette is a so-called soft-sectored diskette, since the start position of each sector is determined by a calculation based on the time interval after the main index hole has passed.

At the beginning of each sector are written a series of identifying marks telling the controller the track and sector is found. It compares these when it has read them with where it has calculated the head to be and sees if there has been an error. Also found at this point at the beginning of each sector is a series of check marks specially encoded to test that the head is decoding the magnetic flux changes properly. Following the above (called preamble) there are 128 bytes of data, followed by some more checking marks called postamble.

Thus the floppy disc controller can randomly retrieve any 128 byte record from the 2002 such blocks on the diskette. All the user has to do is specify the track and sector number where he stored the data, and he can retrieve it years later.

In most actual operating systems, the track and sector numbers are not used by the system user, but files are referenced by a name that is given to them at creation. At the beginning of the diskette there is a cross-reference file which contains the file names and their corresponding track and sector numbers. This file is known as the directory, and is manipulated by the programs of the manufacturer-supplied operating system software to store and later retrieve all files required by the user.

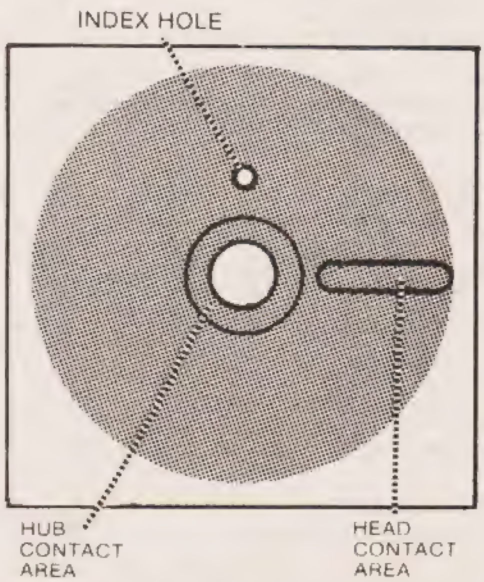


Fig. 3. IBM 3740 Diskette

The exciting new

TRITON

Personal Computer exclusively from:

TRANSAM

Complete kit of parts available only £286 (+ VAT)

Basic in Rom: a powerful 2k Tiny basic resident on board, makes Triton unique, easy to use and versatile.

Graphics: 64 Graphic characters as well as full alpha numerics.

Single Board: Holds up to 8k of memory, 4k RAM and 4k ROM, supplied with 3k ROM and 2k RAM.

Memory Mapping: 2 mode VDU, I/O or memory mapped for animated graphics.

Cassette Interface: crystal controlled Modem tape I/O with auto start/stop + "named" file search.

UHF TV Interface: On board uhf modulator, plugs into TV aerial socket.

Comes Complete with keyboard, case, power supply, thro-hole plated quality P.C.B. full documentation powerful 1k monitor & 2k tiny basic plus all components incl. ic sockets. Note TV set & cassette not included.

EXPANSION: EXPAND YOUR TRITON SIMPLY AND EASILY WITH OUT NEW 8-SLOT MOTHERBOARD — COMPLETE WITH ITS OWN P.S.U. TAKES 8 PLUG IN EUROCARDS. PLUG IN 8K RAM PLUS 8K EPROM CARDS — NOW AVAILABLE.

EXPANSION MOTHERBOARD KIT	£50
8k STATIC RAM BOARD KIT	£97
8k EPROM BOARD KIT	£97

Full details available in our brand new 1979 computer products catalogue – so order your copy now! Don't forget, VAT is extra at 8%. User club £4 per annum.

All components can be bought separately and eleven packs can be purchased on an easy-to-buy scheme. See catalogue.

Triton Humbug V5.1: Runs in 4k on main board. Extended basic includes peek, poke, read, write, call and edit. New 1.5k monitor has software mod for printer plus much more. You return three eproms from Triton and we supply four new ones, fully programmed and documented — **£25** + VAT.

TRANSAM

I am interested in the Triton

Name ____________

Address (please print) ____________

CT2

BARCLAYCARD ACCESS VISA & MASTER CHARGE ARE WELCOME SEND YOUR CARD NUMBER WITH ORDER

To TRANSAM COMPONENTS LTD
12 CHAPEL STREET NW1

Please send me the following

1. A copy of your latest catalogue I enclose 30p + S.A.E 9 × 7 ☐
2. A copy of the Triton Manual I enclose £5 + 70p P & P ☐
3. A complete kit of parts for Triton Computer. £286 + VAT + £4 P & P ☐
4. A Printed Circuit Board £50 + VAT & £1 P & P ☐

TOTAL ENCLOSED £

Cheque, Money Order etc

For details send £1 and SAE.

TRANSAM

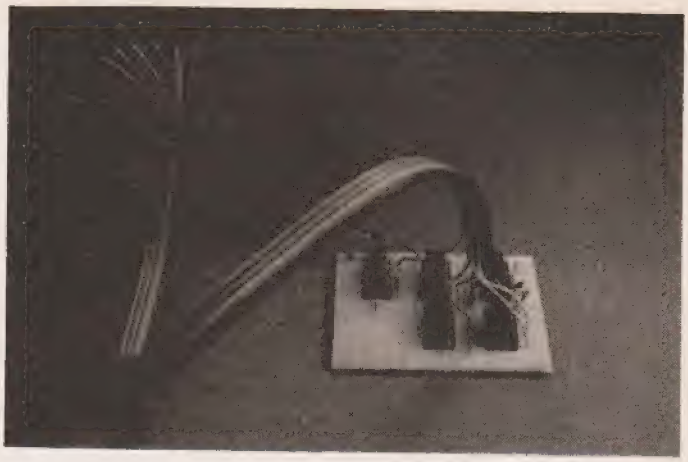
TRANSAM COMPONENTS LTD.
12 CHAPEL STREET
LONDON. NW1 TEL: 402 8137

NEXT TO EDGWARE ROAD TUBE STATION MET LINE
TRITON COMPUTER IS THE TRADE MARK OF TRANSAM COMPONENTS LTD

Mr. D.B. Stiles

PRANG

A random number generator for your games programs

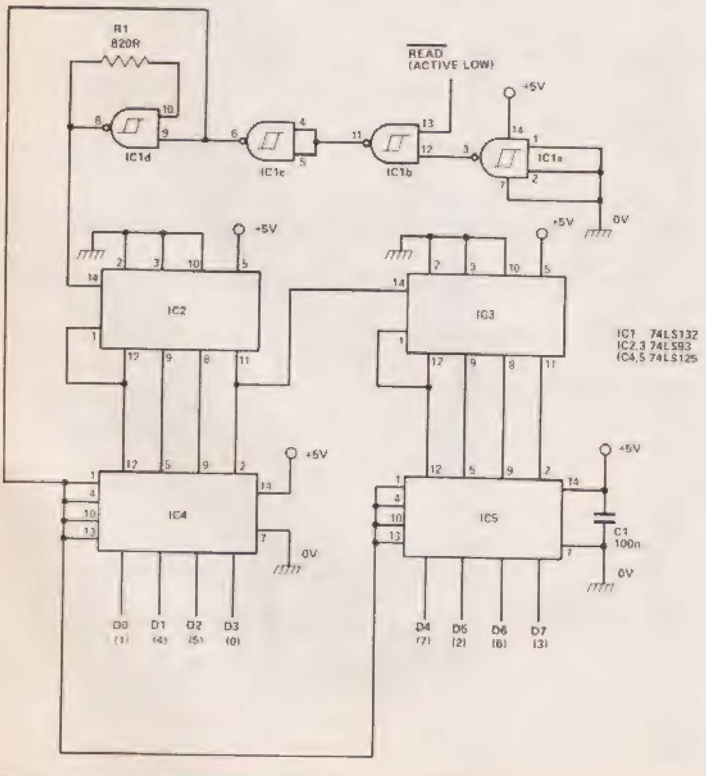


Many games played with the aid of a microprocessor use random numbers, for example, variations on common disc games. The processor is quite capable of generating numbers in a time which the average player would not notice; it can also take a fair proportion of a small systems mem,ory to do it, thus reducing the potential complexity of the games.

The circuit shown here can help reduce the software overhead at the expense of a few chips. It relies for its randomness on the fact that your average micro is quite slow in accessing data compared with the 30 MHZ or so of the clock used for the pseudo-random number generator (PRaNG) shown in the circuit. This, coupled with the asynchronous nature of the two clocks involved, ensure that as far as the processor is concerned the numbers are random. To confuse the micro still further the outputs of the counter relative to the data bus have been randomised! In order to try out the PRaNG, a simple game was devised whereby the infernal machine 'thought' of a number and dared you to guess what it was. Refinements were added to give varying numbers of tries at beating it, (6 or 7 is easy — try 4!), and to make it more easily played by the younger members of the family some prompts were added. Thus came about the 'Guess My Number' program!

The program as given takes about 250 bytes, with some 250 more being taken up with prompting text. The text can be cut down, but not by much. The program is written for Z80 machines, but can fairly easily be re-written for others, as the actual program is fairly simple. Since the input and output programs vary between different machines and monitors these sub-routines have not been included — add your own addresses where applicable. Some systems may also have some of the other sub-routines available, and the DELETE S/R may need to be changed slightly.

Apologies for the odd listing —blame no assembler as 'twas me. One thousand bytes of RAM does not support an assembler these days! The format of the output was designed for use with a Thomson-CSF type VDU, hence the 'Erase Line' commands in the TEXT listing. I must now think of some more number games to justify building the PRaNG. . . . . .



HOW IT WORKS

Prang is assembled from one quad gate package, two binary counters and two quad tri-state buffers. The outputs of these devices can drive TTL in the normal way but also feature a control input which allows the outputs to be switched off so that less than one micro-amp of current flows through them and the device becomes 'invisible' to the data bus. Whilst low-power schottky TTL cannot be considered state-of-the-art the chosen devices combine economy and efficiency in a simple yet effective circuit.

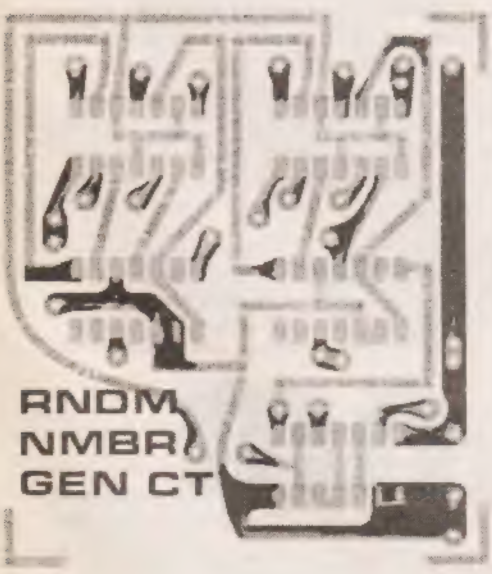
The inputs of IC1a are tied low providing a high level to one input of IC1b whose output is low when the READ input is high. This signal, inverted by IC1c controls the tri-state buffers and the clock generator IC1d which oscillates at about 30 MHz. It is important to use a schmitt device here, a 7400 will not do. The output from IC1d drives the cascaded binary counters IC2 and IC3 which will cycle until the clock is stopped by taking the READ input low, enabling the outputs of IC4 and IC5. When the READ input is taken high again, the outputs of IC4 and IC5 will revert to the tri-state 'off' condition and the clock will restart generating new data ready to be accessed.

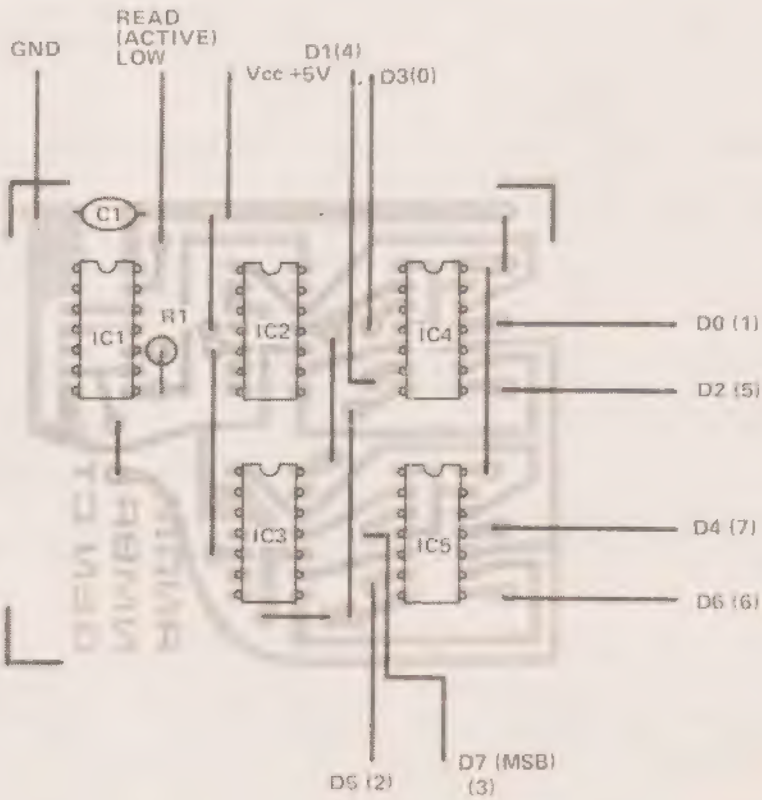
```
* GUESS MY NUMBER: A GUESSING GAME FOR USE WITH
* PSEUDO RANDOM NUMBER GENERATOR ON PORT 02H
* DESIGNED TO USE YOUR INPUT & OUTPUT S/R'S
* 'INCH' RETURNS CHARACTER IN 'A'
* 'OUTCH' EXPECTS AN O/P CHARACTER IN 'A'
                              ORG 1000
1000 START    11 00 00        LD DE, 0
1003          21 00 11        LD HL, TITLE
1006          CD EB 10        CALL OUTS
1009 NEWGAM   21 28 11        LD HL, TRIES
100C          CD EB 10        CALL OUTS
100F          DB 02           IN A, PRNG
*
1011          3C              INC A
1012          27              DAA
1013          47              LD B, A
1014          CD 76 10        CALL INDEC
1017          4F              LD C, A
1018          AF              XOP A
1019          00              NOP
101A          00              NOP
101B NEXT     B7              OR A
101C          3C              INC A
101D          27              DAA
101E          F5              PUSH AF
101F          21 4B 11        LD HL, NUMBER
1022          CD EB 10        CALL OUTS
1025          CD 76 10        CALL INDEC
1028          D8              CP B
1029          20 33           JR NZ, NOTRIT
102B          7A              LD A, D
102C          3C              INC A
102D          27              DAA
102 D         57              LD D, A
102F          21 64 11        LD HL, YES
1032          CD EB 10        CALL OUTS
1035          F1              POP AF
1036          CD C2 10        CALL OUTDEC
1039 END      CD EB 10        CALL OUTS
103C          78              LD A, B
103D          CD C2 10        CALL OUTDEC
1040          21 8C 11        LD HL, SCORE
1043          CD EB 10        CALL OUTS
1046          7A              LD A, D
1047          CD C2 10        CALL OUTDEC
104A          CD EB 10        CALL OUTS
104D          7B              LD A, E
104E          CD C2 10        CALL OUTDEC
1051          CD EB 10        CALL OUTS
1054          CD XX XX        CALL INCH
1057          FE 4E           CP 'N'
1059          20 AE           JR NZ, NEWGAM
105B          C3 XX XX        JP MONTOR
105E NOTRIT   21 BE 11        LD HL, SMALL
1061          38 03           JR C, OUTMES
1063          21 CB 11        LD HL, BIG
1066 OUTMES   CD EB 10        CALL OUTS
1069          F1              POP AF
106A          B9              CP C
106B          20 AE           JR NZ, NEXT
106D          7B              LD A, E
106E          3C              INC A
106F          27              DAA
1070          5F              LD E, A
1071          21 D6 11        LD HL, IWIN
1074          18 C3           JR END
*
* S/R INDEC: INPUT ONE OR TWO DECIMAL NUMERALS TO 'A'
* CONFIRM ENTRY WITH 'CR' : IF YOU CHANGE YOUR MIND,
  'DEL' WORKS
*
1076 INDEC    C5              PUSH BC
1077 WRONG 1  CD XX XX        CALL INCH
107A          CD B3 10        CALL ASCNUM
107D          38 F8           JR C, WRONG 1
107F          47              LD B, A
1080 WRONG 2  CD XX XX        CALL INCH
1083          FE 0D           CP 'CR'
*
1085          20 03           JR NZ, NOTCR
1087          78              LD A, B
1088          C1              POP BC
1089          C9              RET
108A MOTCR    FE 7F           CP 'DEL'
108C          20 05           JR NZ, OK1
108E          CD D9 10        CALL DELETE
1091          18 E4           JR WRONG 1
1093 OK1      CD B3 10        CALL ASCNUM
1096          38 E8           JR C, WRONG 2
1098          4F              LD C, A
1099          78              LD A, B
109A          OF              RRCA
109B          OF              RRCA
109C          OF              RRCA
109D          OF              RRCA
109E          B1              OR C
109F          4F              LD C, A
10A0 WRONG 3  CD XX XX        CALL INCH
10A3          FE 7F           CP 'DEL'
10A5          20 05           JR NZ, OK2
10A7          CD D9 10        CALL DELETE
10AA          18 D4           JR WRONG 2
10AC OK2      FE OD           CP 'CR'
10AE          20 FO           JR NZ, WRONG 3
10B0          79              LD A, c
10B1          C1              POP BC
10B2          C9              RET
*
* S/R ASCNUM: CONVERT CONTENTS OF 'A' TO BCD IS ASCII
  CODE IS VALID
* IF IT'S NOT, CARRY FLAG IS SET
* IF CORRECT, NUMBER IS RE-CONVERTED TO ASCII & O/P TO TTY
*
10B3 ASCNUM   FE 3A           CP 3AH
10B5          30 09           JR NC, NOTNUM
10B7          D6 30           SUB 30H
10B9          38 05           JR C, NOTNUM
10BB          CD CF 10        CALL OUTNUM
10BE          B7              OR A
10BF          C9              RET
10C0          C9
10C0 NOTNUM   37              SCF
10C1          C9              RET
*
*
* S/R OUTDEC & OUTNUM: CONVERT CONTENTS OF 'A' TO
  ASCII (0-9)
* AND O/P TO TTY, HIGH NIBBLE FIRST: SUPPRESSES LEADING
  ZERO
* 'OUTCH' IS YOUR S/R!!
*
10C2 OUTDEC   F5              PUSH AF
10C3          E6 FO           AND FOH
10C5          28 07           JR Z, LODEC
10C7          OF              RRCA
10C8          OF              RRCA
10C9          OF              RRCA
10CA          OF              RRCA
10CB          CD CF 10        CALL OUTNUM
10CE LODEC    F1              POP AF
10CF OUTNUM   F5              PUSH AF
10DO          E6 OF           AND OFH
10D2          F6 30           OR 30H
10D4          CD XX XX        CALL OUTCH
10D7          F1              POP AF
10D8          C9              RET
```

```
* S/R DELETE: REMOVE CHARACTER FROM VDU SCREEN
  (STAND ALONE TYPE)
* AND REPOSITION CURSOR OVER DELETED CHARACTER
* FOR OTHER TYPES OF VDU, ALTER ACCORDINGLY!
* S/R OUTCH IS YOURS!
*
10D9 DELETE    F5          PUSH AF
10DA           3E 08       LD A,   'BS'
10DC           CD XX XX    CALL OUTCH
10DF           3E 20       LD A,   'SP'
10E1           CD XX XX    CALL OUTCH
10E4           3E 08       LD A,   'BS'
10E6           CD XX XX    CALL OUTCH
10E9           F1          POP AF
10EA           C9          RET
*
* S/R OUTS: OUTPUT A N ASCII STRING , START ADDRESS-1
  IN 'HL'
0 STRING TERMINATOR IS 'ESC' (1BH) OR ANY CHARACTER
  YOU DESIGNATE!
*
10EB           OUTS
10EB OUTS      23          INC HL
10EC           7E          LD A,(HL)
10ED           PE 1B       CP    'DSC'
10EP           C8          RET Z
10F0           CD XX XX    CALL OUTCH
10F3           18 F6       JR OUTS
*
*
* TEXT STRINGS: ITEMS SHOWN THUS:- (CR) ARE CONTROL
  CHARACTERS
* (CR) CARRIAGE RETURN OD HEX
* (LF) LINE FEED   OA HEX
% (ERL) ERASE LINE (SUB) 1A HEX USED ON THOMSON-CSF
  TYPE VDU'S
* (ESC) STRING TERMINATOR 1B OR ANY OTHER OF YOUR
  CHOICE!
* SPACES ARE SHOWN BY UNDERBAR THUS: -

                 ORG 1100
1100   (ESC) GUESS§MY§NUMBER;§IT'S§BETWEEN§O§AND§99.
       (ESC)
1129   (CR) (LF) (ERL) HWOW§MANY§TRIES§WOULD§YOU§LIKE?§
       (ESC)
114C   (CR) (LF) (ERL) WHAT§IS§YOUR§NUMBER?§(ESC)
1165   (CR) (LF) (ERL) CLEVER!§IT§TOOK§YOU§(ESC)
117D   §GOES§TO§GUESS§(ESC)
118D   (CR) (LF) (ERL) THHE§SCORE§IS§YOU§(ESC)
11A2   §ME§(ESC)
11A7   (CR) (LF) (ERL) PLAY§ANOTHER?§Y§OR§NO (ESC)
11BF   §§TOO§SMALL; (ESC)
11CC   §§TOO§BIG; (ESC)
11D7   (CR) (LF) (ERL)I§WIN!!§IT§WAS§(ESC)
```

PARTS LIST

C1	100 n
R1	820 R
IC1	74LS132
IC2,3	74LS93
IC4,5	74LS125





Mr. A.P. Stephenson

D2 PROGRAMMING

A series of articles on programming techniques for the Motorola D2

The PROGRAMS are intended to guide the poor student through the mysteries of machine language programming using the MOTOROLA MEK D2 evaluation kit. The PROGRAMS commence at elementary (perhaps trivial) level and progress slowly towards the impossible. The format of the programs include the ASSEMBLY LANGUAGE equivalemt. At first sight, this would appear to be a waste of time because the D2 kit can only recognise programs written in hexadecimal machine code. Nevertheless, it is a good habit to write programs in ASSEMBLY LANGUAGE first because of the superior memonic aids provided by the language. It also helps in tracing bugs.

It must be pointed out however that full ASSEMBLY tricks are not included. For example, the operands are given in hexadecimal absolute address instead of the usual symbolic form in order that ıprograms appear free of the clutter resulting from DECLARATION lines at the top. It is unfortunate that symbolic addresses, although intended to ease the burden of programming, often confuse the student (and the writer) by the inclusion of this mass of necessary jargon before the actual program begins.

Hardware Needs

The programs that refer to PIA input and output assume that the D2 kit has been extended to include lamps and switches. Although simple switches are normally good enough for the sixteen data terminals of the PIA, it is advisable to employ de-bouncing flip-flops on the CA1, CA2, CB1 and CB2 switches. If this is not done, the operator must expect that active HIGH and active LOW transitions of the switches may give undefined results; for example, if the control register is say, programmed to cause interrupt when CA1 goes from HIGH to LOW, then it is quite on the cards that a LOW to HIGH transition will also cause interrupt due to the switch trembling.

The first line in all programs is LDS 00FF which sets the stack pointer at the last address (00FF) in the users RAM. Unless this is done, the executive program in ROM called JBUG will tend to corrupt the programs. There is no rule that states that the stack pointer MUST be set to address 00FF but it must be set somewhere so the safest place is in the last address. If however, your income entitles you to belong to the well-spliced set, then you have probably bought another 256 bytes of RAM in which case YOUR last address is elevated to 01FF.

Manually Assisted

The literature supplied with the D2 Kit includes a comprehensive PROGRAMMIG MANUALL which devotes an entire page to each instruction. Unfortunately, it does not attempt to explain how to program — it is assumed you already know this.

It is unfortunate that standard works on the subject tend to give example programs which, far from helping the beginner, only demoralise him or her. Rather than wade through masses of prelimiunary work, the best way to learn programming is to study carefully every line of the following examples and then rewrite it with some slight variation, perhaps a change of address or relocate it in another block of memory.

Try to develop a critical attitude towards them and try to save the odd byte by a better choice from the instruction repertoire. If the program does something to the PIA A-side, modify it to the B side or vice versa. If any line used EXTENDED' addressing when the more economical DIRECT addressing could have been used then modify it — but remember that if the program used BRANCH instructions, the RELATIVE address may require channging.

As early as possible, try to write your own programs but keep them simple because frustration in the early stages could lead to a violent physical attack upon the keyboard. Watch out for finger trouble at the keyboard and keep a watchful eye onm that "Prompt" bar on the left hand display digit. This MUST be showing before any fresh execution run or change of JBUG function. When in doubt, press E and start again is a good rule of thumb.

When entering a program don't forget to press the M button after keying in the first address; it is rather annoying after entering about thirty or more byte of code to discover that none of it has gone in! Be very methodical when entering any program. Don't rush; pause after EVERY instruction has been entered and check the address digits on the display correspond with what the program declares they should be.

High To Low Level

This simple little trick will ensure you don't miss the odd byte out; remember that a missed byte in the first few instructions means that almost the entire program has to be re-entered. If you have been used to the luxury of BASIC, whereby the missed instruction can be slipped in at the bottom, these pitfalls ill come as an unpleasant shock. In fact, those who arrive at microprocessors after an apprenticeship in BASIC are in danger of disillusionment.

Programmin g in hexadecimal machine language demands an intimate knowledge of the INSTRUCTION CODES, an ability to think at machine level (which is very primitive) and above all, patience and fortitude. Nevertheless, the skill is a rewarding one, the overall understanding of computerisedintelligence can only be gained by using the language of the creature; high level languages, although ideally suited for using a computer will never allow the user to see through the mountain of software between him and the machine.

He is conversing through an interpreter and as in normal life, a translation can never equal the original in subtlety of phrasing. Because the forgoing is to some extent controversial and may be taken as disparaging to the exponents of high-level language, it may be useful to set out the relative advantage and disadvantages of machine and high-level language.

Machine Language

a) The machine codes must be understood. Some knowledge of the machine hardware necessary
b) The code, once understood, is ONLY APPLICABLE TO THAT PARTICULAR COMPUTER.
c) The various ADDRESSING modes applicable to each instruction code must be understood.
d) Every single step of a complex program must be set out in tortuous detail.

e) Calculating the number of bytes to jump back or forward in BRANCH instructions is time consuming and error prone.
f) The final program can be executed immediately and (providing it is written sensibly) is efficient, fast and will be economic in terms of memory capacity.

High-Level Language
a) Easy to learn.
b) Knowledge of the machine codes and the various addressing modes is not required.
c) Knowledge oif the machine hardware is unnecessary.
d) A program written for one computer will also run (perhaps after some modification) on any other computer PROVIDING you have a TRANSLATION program (written for that computer). The translation program for BASIC is called an INTERPRETER; the translation program for FORTRAN, ALGOL, CORAL, etc, is called a COMPILER.
e) The compiler languages operate as follows:
  i. The COMPILER must be resident in ROM (or perhaps floppy disk)
  ii. The program written in high-level (called the SOURCE program) is them COMPILED(translated) within RAM into a set of machine language instructions called the OBJECT PROGRAM:
  iii. The machine language object is now EXECUTED.
f) An interpreter language such as BASIC operates as follows:
  i. The INTERPRETER must be resident in ROM (or perhps floppy disk)
  ii. THE SOURCE program is translated line by line and EXECUTED line by line which means that the INTERPRETER occupies RAM memory during the EXECUTOION.
g) Each line (called a STATEMENT) of a high-level source program may cause the execution of many machine language steps. In general, the number of machine language steps will be greater after translation than would have been the case if the program had been written directly in machine language. High level language is therefore less efficient in terms of speed of execution. It is also more greedy in memory requirements.

For example a BASIC interpreter requires between four and ten kilobytes of storage space, dependent on the quality and facilites offered. For example, we are inundated with various hybrid forms of the original "Dartmouth" BASIC due to the explosion of the microprocessor-based "home computer". Thus we have "micro BASIC" (bare minimum), "mini BASIC" (bit better), BASIC (defined in the middle sixties at Dartmouth College in the USA) and a form called "extended BASIC" which is alledged to offer more powerful facilities than the original.

Assembly Language
Where does Assembly Language fit in? As described under compiler and interpreter languages, some form of translator program is still required if programs are to be writen in ASSEMBLY LANGUAGE. There is however a fundemental difference.

A program written in assembly language has a one-to-one correspondence with the machine language translation. In fact it may be described as "memonically coded machine language". The Operation codes are replaced by three-letter groups having some meaning. The addresses (OPERANDS) can be absolute addresses or, if the programmer chooses, can be letter groups off his own choice providing the absolute address associated with the group is stated at the beginning of the program (called the declaration or DIRECTIVE). Any instruction to which a BRANCH is subsequently made can be given a letter group LABEL (of the programmer's own choice).

Thus the operand of a Branch instruction need not be laboriously calculated by counting the bytes — all that is required is the label. In spite of this, it must not be thoiught that assembly language is easy. In fact the programmer must still think at machine level, it is still necessary to have some knowledge of the machine hardware, it is still necessary to understand the various addressing modes so it is still as difficult to write a program. There is however an important advantage.

Assembly language is far less error prone than hexadecimal machine language because of the powerful mnemonic aids of the letter groups and the BRANCH LABELS. In addition, some assemblers have built in editing aids. One final point is worth noting — a hexadecimal keyboard is cheap, but a full ASCII coded keyboard isn't and one of these, plus a few kilobytes of ROM must be purchased if you wish to up-grade your installation for assembly language programming facilities. And of course, you need the assembler program inside the ROM! By the way, if you ever become elevated tó the peerage and decide to buy a PET 2001 or some similar "Mainframe giant," don't stick permanently on BASIC. Have a go with the excellent Assembler provided and get to know the microprocessor behind the glossy exterior. In short, don't remain a BASIC bumpkin all your life, learn the delights of intellectual frustration with an Assembler.

Format Of The Example Programs
The first colmn is simply the LINE NUMBER and proceeds in decimal sequence. The second column is the hexa decimal address where the FIRST BYTE of each instruction is stored in RAM. Remember here that memory is only ONE BYTE WIDE so each address can only hold TWO HEX CHARACTERS: This means that instructions which are three bytes long burn up three addresses which accounts for the rather jumpy progression of this column.

The third column is the MACHINE LANGUAGE PROGRAM. The first two characters represent the OPERATION CODE part of the instruction, ie, it tells the computer WHAT it is required to do. The remaining characters, if any, inform the computer what the data is or WHERE in memory it can be found.

Some instructions however, by their nature, do not require further information). It must be understood that column three is the complete MACHINE LANGUAGE PROGRAM
program and the only column that can be entered by a hexadecimal keyboard into RAM. The remaining columns, four, five, six and seven belong to the program written in ASSEMBLY language and strictly, are not required even to be writen if the MEK D2 kit is to be used.

Nevertheless, it is advised that, even though it has no meaning to the machine, the Assembly columns should be included. In fact when first writing a program, this should be the first task after the preliminary flow chart has been scribbled because:
a) it is easier the "think" in mnemonic letter groups
b) if the program doesn't work first time (and it probably

won't) it is easier to find bugs in the logic flow.

c) if you eventually up-grade to an assembler version of the kit, you will have had at least a background feel for the new methods of writing programs.

d) other peoples programs written in magazines or manufacturers literature are always presented in both Assembly and Machine language; in fact, a copy of the so-called "Assembly Listing" outputted from the machine. It is worth emphasising again at this point that the example pograms which follow are not completely representative of strict Assembly language because the Operands (addresses) are left in ABSOLUTE form in hex rather than symbolic letter groups in order to reate directly with the machine language lines.

Column four is the LABEL field and will only be used if the program has lines to which BRANCHES or JUMPS are to be made. The labels are chosen arbitrary by the programmer. Column five is the MNEMONIC OPERATION CODE. Column six is the ACCUMULATOR field, used to distinguish between ACC "A" and ACC "B". If this instruction does not refer to an accumulator at all then of course it would be left blank.

Column seven is the OPERAND field, which refers to the data or where abouts in memory the data can be found. Because there are several possible interpretations of an operand, defined symbols must be used to "tell" the assembler which addressing mode is intended. The symbolism used in the examples takes the following form:

a) DIRECT ADDRESSING____two hexadecimal characters

b) EXTENDED ADDRESSING__four hexadecimal characters

c) IMMEDIATE ADDRESSINGthe symbol preceeds the hexadecimal characters

d) INDEXED ADDRESSING__two hexadecimal digits follwed by a COMMA followed by X

The official MOTOROLA assembler allows the operand characters to be assumed decimal unless preceded by the $ sign, in which case they would be assumed Hexadecimal. Since the D2 kit inputs only hexadecimal characters, this sign would be appearing before every address so for purposes of simplicity it has been omitted. Column eight is the REMARK which the programmer hopes may give some insight into his schizophrenic reasoning.

Analysis Of Example Programs

To understand the programs, the following information is assumed to be at hand:

a) The set INSTRUCTION CODES for the MOTOROLA 6800 micropressor

b) The diagram depicting the function of each bit in the PIA Control Register.

Fig. 1. Board layout:

The kit consists of two boards, interconnected by a 50 way ribbon cable:

BOARD "A" contains the microprocesor, clock,PIA(dedicated to the keyboard), PIA for the user, ROM chips containing the " JBUG " monitor program, RAM for JBUG, RAM for user and a few logic chips.

BOARD "B" contains the hexadecimal keyboard, six digital display characters, ASIA dedicated to interfacing the tape recorder "backing store" and some logic chips.

RAM And ROM Capabilities:

JBUG______1K ROM organised on the address bus at E000 to E3FF
JBUG "scratch-pad" RAM— — — " " A000 to A07F

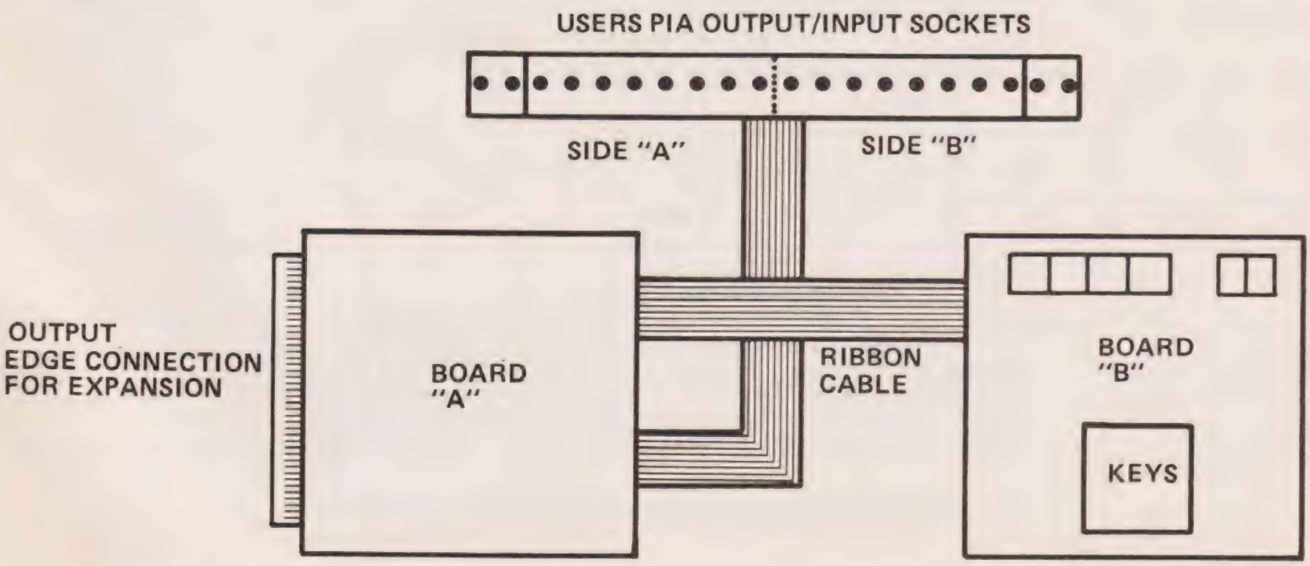
Users RAM (256 bytes)______ " " 0000 to 00FF
Sockets are provided on board "A" to increse users RAM by another 256 bytes and users ROM of 256 bytes.

Special Addresses:

User PIA SIDE "A" — — — — — — — — — — — — — 8004 and 8005
User PIA SIDE "B" — — — — — — — — — — — — — 8006 and 8007

AUDIO CASSETTE
Load starting address of block in A002 and A003
Load finishing address of block in A004 and A005

Interrupt vector addresses — — — — — — — — — — — — — A000 and A001



ASSEMBLY LANGUAGE

1	0000	8E 00 FF		LDS		00FF	Set up stack at 00FF
2	3	96 26		LDA	A	26	Load 1st number
3	5	9B 27		ADD	A	27	Add 2nd number
4	7	97 28		STA	A	28	Store
5	9	3F		SWI			Stop
6							

REMARKS
OPERAND
ACCUMULATOR
MNEMONIC OP-CODE
LABEL FIELD

OPERAND
HEX MACHINE CODE
OP-CODE

INSTRUCTION ADDRESS

LINE NUMBER

Note: 1. The program is trivial and simply adds contents of addresses 26 and 27 and stores sum in 28

2. Always write out programs in above format

3. The symbol denotes the operand is IMMEDIATE

4. Hex addresses should strictly be preceeded by the symbols but in these examples they will be omitted.

Program 1

Line 1— sets the STACK-POINTER at the last address in user's RAM. Note that the addressing mode is IMMEDIATE because OOFF is to be considered **data** (not the address of where data is). Note also that, because the Stack Pointer is 16 bits long, a full four-bit address is required; thus FF won't do — it must be written as OOFF.

Line 2 — This LOADS the content of address 26 into ACC A, using the machine code 96 which is **direct** addressing. Note carefully that we use Load (LDA) when going FROM a memory location TO a register. It is also relevent to stress at this point that after LDA, the original data in the memory location is STILL there but the Accumulator will now have its old contents overwritten by the new data. This is why the first number was LOADED into the Accumulator in order that previous "garbage" is destroyed.

Line 3 — This adds the contents of address 27 to the previous contents of ACC A. Again, the original contents of address 27 remains unharmed.

Line 4 — Stores the contents of ACC in the address 28; note that we use STORE (STA) when going FROM a register TO a memory address. This time however, the ACC still retains its data but the memory location now holds the new information: the old contents are overwritten.

Line 5 — stops the program. There is no actual STOP instruction in the repertoire but SWI (Software Interrupt) has the same effect with the additional advantage that all the registers have their contents copied into the STACK enabling them to be examined with the aid of the Register Examine mode in JBUG.

Note carefully the progression in column two (the memory addresses where the instructions are stored). The first byte of the first instruction is stored in address 0000 (an arbitrary but logical choice for the beginning address). Because this instruction is three bytes long, the beginning of the next instruction is placed at address 0003. This is two bytes long so the first byte of the next instruction is at address 0005 — and so on.

The discussion of this elementary (perhaps trivial) program has taken up a lot of space but it is essential at the beginning of one's programming career to grasp the simple well — the difficult programs are then easy! Such detail in the remaining programs will not be given unless a new instruction or idea is employed.

SWOP CONTENTS OF A AND B

1	0000	8E 00 FF		LDS		00FF	
2	3	D7 10		STA	B	10	
3	5	16		TAB			
4	6	96 10		LDA	A	10	
5	8	3F		SWI			

PROGRAM 2

This program illustrates how useful it is to have more than one Accumulator (some microprocessors only have one, some have eight or more). Remember that the program assumes that addresses 26 and 27 already contain "Data". To test if the program works, it is of course necessary to previously place some data in these locations using the "M" button. So the rule for testing out all programs, such as this one and the previous one, is as follows:

If the program assumes data somewhere, ENTER THIS DATA BEFORE the program is RUN using the "M" button.

D2 PROGRAMMING

If this is not done, there is no way of telling if the program is free of bugs. It may be worth mentioning here that when entering arbitrary data in order to test a program it is better to choose rather small numbers if the program is intended to perform arithmetic. Thus if two numbers are to be added, pick numbers like 03 and 05 so the answer is not into the hexadecimal letters. Thus if you pick 29 and 54 then the sum should come to 7D (which may not be immediately obvious unless you and the microprocessor both have hexadecimal brains).

SWOP CONTENTS OF 26 AND 27							
1	0000	8E 00 FF		LDS		00FF	
2	3	96 26		LDA	A	27	
3	5	D6 27		LDA	B	27	
4	7	97 27		STA	A	27	
5	9	D7 26		STA	B	26	
6	B	3F		SWI			

PROGRAM 3

This is similar to Program 2 but swops the two accumulators over (or rather the contents of them). Note the very economical instruction TAB meaning "transfer contents of ACC A to ACC B" (thus they will both have the same data as was in ACC A before the instruction). Note that TAB uses INHERENT addressing so an address byte is not required. One snag here, how can we place data in the two accumulators to test the program out before it is run? This is where old JBUG comes to the rescue because after line 5, which is SWI, all the registers are stored methodically into the STACK. Since we have set the BOTTOM of the stack at OOFF, the old chap stores the PROGRAM COUNTER (PC) in the addresses OOFF and OOFE, the INDEX REGISTER (X) in the addresses OOFD and OOFC, ACC A in OOFB and ACC B in OOFA, etc. Thus to place test data in ACC A and B, use the "M" button to expose the address OOFA and OOFB. It is essential to execute a dummy run first in order that SWI can first do its job; then place in the data before you run it again.

ADD 1 TO A, ADD 3 TO B, ADD 1 TO CONTENTS OF 27							
1	0000	8E 00 FF		LDS		00FF	
2	3	4C		INC	A		Add 1 to A
3	4	CB 03		ADD	B	03	Add 3 to B
4	6	7C 00 27		INC		0027	Add 1 to contents of 27
5	9	3F		SWI			

PROGRAM 4

The simple way to "add 1" to an accumulator or a memory location is to use the instruction INC which means "increment". Line 2 uses the code 4C which increments ACC A. Line 3 uses IMMEDIATE addressing to add 3 to ACC B. Line 4 uses INC again to add 1 to the contents of address 27. Slight query here, why was the EXTENDED address used requiring 0027? Since address 27 is supposed to be in the bounds of DIRECT addressing, why didn't we use this? The answer is that the instruction INC doesn't allow direct addressing, otherwise it would have been used.

ADD 5 TO CONTENTS OF 26 AND CLEAR A000							
1	0000	8E 00 FF		LDS		00FF	
2	3	96 26		LDA	A	26	
3	5	8B 05		ADD	A	05	
4	7	97 26		STA	A	26	
5	9	7F A0 00		CLR		A000	
6	C	3F		SWI			

PROGRAM 5

Line 3 adds the 5 using IMMEDIATE addressing with the ADD instruction.

Line 5 "clears" (fills with Zeros) the contents of address AOOO by using CLR with EXTENDED ADDRESSING (because it is a four hexadecimal address).

MINI-ADS &

TRS-80 £620.00

We have no expensive consultancy or shop front overheads - - only low prices.

Tandy TRS-80 16k Level II

Complete with CPU/ keyboard, Video monitor, Cassette recorder, Manuals and new numeric keyboard. £620.00 + VAT

240 volt power supply £10.00

Delivery within 14 days of cheque clearing (mainland only).

MICROCOMPUTER IMPORTS
Tel. (0276) 24291
Mail order address
129 Upper Chobham Road
Camberley, Surrey

SUPERBOARD II

4K — £263* 8K — £299*

(modulator included)

FIBRE GLASS CASE £24.95

TRS 80 KEYBOARD

4K LEVEL II — £390* 16K — £490*

(modulator included

POWER SUPPLY £9.90*

ATARI VIDEO COMPUTER

VCS — £139* CARTRIDGES £12.95*

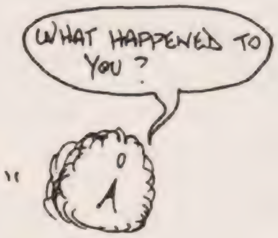
NEW — CODE BREAKER BRAIN GAMES
HUNT & SCORE

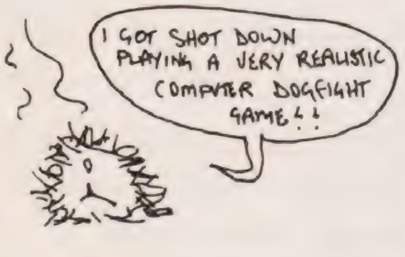
SOFTY EPROM DEVELOPMENT-BURNER SYSTEM

(kit) — £79.50*

* PRICES EXCLUDE 8% VAT — FULL DETAILS ON REQUEST

VIDEOTIME PRODUCTS
56 Queens Road, Basingstoke,
Hants. RG21 1REE
Tel. (0256) 56417. Tlx. 858747





The Gospel according to E.T.I.

PET AND TRS80 PROGRAMS URGENTLY REQUIRED by established software company, for nationwide sale. High royalty payments for top-quality original software — any application. Phone Chris Holbrook on 0222-372066 anytime.

NASCOM in nice cabinet with heavy duty fan cooled protected P.S.U., CUTS interface, B-BUG, cassette and all data books. £250. 8K dynamic RAM £40. Bonner, Orpington 26802.

6800 ASSEMBLER in 2KPROM, processes additional 6801 instructions, position independent, no extra RAM required, MINI/ MIKBUG compatible. Also available super TTY monitor. SAE for details Leisure Time Ltd., Crozier Road, Mutley Plymouth.

FOR SALE, a basic ELF II microcomputer. Constructed, cased and operational, complete with expansion power supply and documentation. £170 o.n.o., C. Goudie, 34 Crofthead Avenue, Kilmarnock KA3 2HA.

TRACTOR PRINTER FOR SALE, complete with Keyboard. Also available a paper tape reader/punch and winders. Can be delivered. Offers. Phone Crawley 510358 after 5 p.m.

WANTED: TRS-80 Level-1 4K microcomputer. Will pay up to £250 (negotiatable). Telephone 061 795 5025 after 5.30. Manchester.

NASCOM 1, microcomputer, PSU, cased and working. T2 monitor and documentation. £210 ono. Alan Knifton, Derwent Cottage, Sitch Lane, Oker, Matlock, Derbyshire (0629 83 4426).

NASCOM 1 system with PSU and B-Bug, wired, and tested by Nascom. £200 complete. Will deliver within reason. Derby 880526 evenings.

NASCOM 1, B-Bug, buffer, cassette recorder, games tape, documentation. Exchange for anything to produce hard copy from a "Pet" or sell for best offer. Stanton (Suffolk) 0359-50619.

TRS 80 SOFTWARE. Full business systems, bookkeeping, mailing, text writing. Write for details and advice BOX 115, 30 Heathfield Road, Croydon, Surrey.

TRS 80 HARDWARE. Also lower case printers, micropolis disc drives. BOX 114, 30 Heathfield Road, Croydon, Surrey.

CLASSIFIED

AD INDEX

THIS SECTION IS A PRE-PAYMENT SERVICE ONLY

MINI-ADS: 31/4" x 21/8", 1-3 £38, 4-11 £36, 12 or more £34 per insertion. CLASSIFIED DISPLAY: 19p per word. Minimum 25 words. Boxed classifieds are £6.33 per col. centimetre. No P.O. Box Numbers can be accepted without full address.

Enquiries to: Advertising Department, 01-437 5982. 25-27 Oxford Street, London W1R 1RF

INTERESTED IN HOME COMPUTING?

FREE B BUG
valued at £23.00
plus 10 x C12 cassettes
valued at £4.00
plus Standard Modulator
valued at £2.25
WITH EVERY NASCOM

Start now and don't get left behind THE NASCOM 1 is here Ex-stock with full technical services

Plus the opportunity to join the fastest moving club of personal computer users enabling you to get the most our of your computer. You can OBTAIN and EXCHANGE programs and other software — many now available.

SAVE £60

The Powerful Z80 Microprocessor
Professional Keyboard
1 Kbyte Monitor in EPROM
2 Kbyte RAM (expandable)
Audio Cassette interface
Plugs into your domestic TV
Easy construction from straightforward instructions
— no drilling or special tools
— Just neat soldering required.

Only ~~£197.50~~ + 8% VAT (includes p & p + insurance)

NEW LOW PRICE £165 + VAT

Manuals seperately	2.95
Z80 programming Manual	6.90
Z80 Technical Manual	2.95
PIO Technical Manual	2.95
(All prices add 8% VAT)	

Power supply suitable for NASCOM 19.90

NASCOM AD ONS — Nascom improved monitor B Bug (2K) featuring — *Four times tape speed *Direct text entry without ASCII *Extended keyboard facility *Additional useful subroutines **£23.00**

Nascom Vero Case **£22.50**

Nascom Joy Stick Kit **£14.90**

Nascom Music Box Kit **£9.90** *(write your own tunes and play them on your Nascom. Complete with full documentation).*

GRAPHICS ADD ON BOARD £9.90
Complete kit to upgrade your NASCOM for graphics capability includes full documentation and demonstration program.

NASCOM IMMEDIATE EXPANSION S100 from COMP —strongly recommended

The only available S100 motherboard kit (fully buffered) that plugs directly into your Nascom. Designed for the insertion of S100 boards (e.g. Static RAM, EPROM and discs etc.).

S100 Motherboard/Buffer **£47.50**
(Complete kit + documentation)
Suitable 8K Static RAM Memory ~~£125~~ **£110**
(fully assembled tested and guaranteed)

Motherboard comes complete with 2K Tiny Basic On cassette.

NASCOM LOW COST EXPANSION

Uses dynamic RAM and NASBUS
(please note this expansion does not support S100 memory)

Tiny Basic in EPROM **£25.00**
8K Dynamic RAM board (in kit only) **£85.00**
Motherboard (in kit only) **£12.50**
Buffer board (in kit only) **£25.00**
All plus VAT

SHORT C12 CASSETTES 10 for £4.00

FOR COMPUTER PROGRAMMES

SPECIAL DYNAMIC MEMORY OFFER

£69 + VAT

8 x 4116 □ 150ns access time, 320ns cycle

COMPUKIT UK101 —ON DEMO

All specifications of Superboard with many improvements and in kit form including power supply and modulator on board. Improved video display with 48 chars x 16 lines. English TV Standard.

£219 + VAT

RELEASED END OF JUNE 1979.

SEND £10 DEPOSIT TO RESERVE

OHIO SUPERBOARD II *NEW* FIRST IN UK —ON DEMO

For electronic buffs. Fully assembled and tested. Requires +5V at 3 Amps and a video monitor or TV with RF converter to be up and running.

STANDARD FEATURES
Uses the ultra powerful 6502 microprocessor
8K Microsoft BASIC-in-ROM
Full feature BASIC runs faster than currently available personal computers and all 8080-based business computers.
4K static RAM on board expandable to 8K
Full 53-keyboard with upper/lower case and user programmability
Kansas City standard audio cassette interface for high reliability
Full machine code monitor and I/O utilities in ROM

Direct access video display has 1K of dedicated memory (besides 4K user memory), features upper case, lower case, graphics and gaming characters for an effective screen resolution of up to 256 by 256 points. Normal TV's with overscan display about 24 rows of 24 characters; without overscan up to 30 x 30 characters.

EX-STOCK — £263 + VAT

Please phone to check availability or send £10 to reserve one and pay the balance on delivery

MODULATORS UHF Channel 36

Standard 6 meg band width **£2.25**
High Quality 8 meg band width £4.90

INDUSTRIAL POWER SUPPLY

+5v 10 amps — **£63.25** +5v 5 amps +12v-12v-5v **£89.00**
DC switched Both housed and fully guaranteed. Lightweight

TRS 80 SOFTWARE *NEW*

100 MIXED PROGRAMMES on cassette **£39.00** (inc.VAT)

HITACHI 9" & 12" PROFESSIONAL MONITORS

SUPERB DEFINITION

9" — **£132**
12" — **£210**
+ VAT

Delivery charged at cost — Red Star — Securicor — Post etc., Your Choice. Please make cheques and postal orders ayable to COMP, or phone your order quoting BARCLAYCARD or ACCESS number.

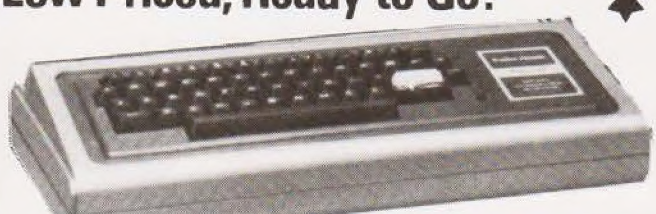
OPEN — 10am to 7pm — Monday to Saturday
CONTINUOUS DEMONSTRATIONS

SEND FOR OUR SPRING '79 CATALOGUE - 0.60p

JUST COMPARE OUR CASH AND CARRY PRICES!

SEND FOR OUR SPRING '79 CATALOGUE - 0·60p

THE TRS-80 *(SPECIAL SCOOP)*
Low Priced, Ready to Go!



Use your own cassette

Level-II with 4K RAM
Improved graphics, print formatting, and a faster cassette transfer rate are features of Level-II BASIC.

£399 + VAT

PLUGS INTO YOUR OWN TV

Level-II with 16K RAM
A combination of 16K RAM and the powerful Level-II BASIC produces a system capable of handling most demands.

£499 + VAT

FREE 100 PROGRAMS

KEY BOARD ONLY | COMPLETE WITH UHF MODULATOR

UK POWER SUPPLY — **£9.90** + VAT

ATTENTION! TRS 80 & APPLE II USERS

SAVE £90

Simple to fit - only a screw-driver is required.

16K UP GRADE KIT — HALF NORMAL PRICE — **£99** + VAT — LIFETIME GUARANTEE

NEW Standard QWERTY Keyboard with 16K Memory

only £625 + VAT

The No. 1 in the U.K. Affordable

8K PET still at unbeatable price of **£499** + VAT

NEW! THE ITT APPLE (2020)

EX STOCK 4K
LOW PRICE MEMORY UPGRADE
PURCHASED AT TIME OF ORDER

16K £49 + VAT

* Full colour - UHF output * Audio cassette tape interface * Up to 48K RAM on board * BASIC in ROM (graphics commands include COLOUR = VLIN, HLIN, PLOT and SCRN) * Built in Loudspeaker *Buckets of software already available *disk system (110 K byte per drive — includes controller) only £425 + VAT

THE EXIDY SORCERER.

SAVE

SORCERER COMPUTER SYSTEM

CPM NOW

The Sorcerer Computer is a completely assembled and tested computer system. Standard configuration includes 63-key typewriter-style keyboard and 16-key numeric pad, Z80 processor, dual cassette I/O with remote computer control at 300 and 1200 baud data rates, RS232 serial I/O for communications, parallel port for direct Centronics printer attachment, 4K ROM operating system, 8K ROM Microsoft BASIC in Rom Pac™, cartridge, composite video of 64 char/line 30 line/screen, 128 upper/lower case ASCII set and 128 user-defined graphic symbols, operation manual, BASIC programming manual and cassette/video cables, connection for S-100 bus expansion.

LOOK!
*32K RAM on board
*RS232 interface *8K BASIC ROM
*CUTS interface *4K MONITOR
*KANSAS CITY interface *S100 BUS
*User defined graphic symbols *Z80 cpu

EXIDY SORCERER 32 RAM COMPUTER **£790** + VAT

BOOK SHELF

LOW PRICES FAST SERVICE!

Vol 0 The Beginner's Book	£5.40
Vol 1 Basic Concepts +	£5.90
Vol 2 Some Real Products (June 1977)	£9.90
Microprocessor series, by Rodney Zaks C201 From Chips to Systems	£7.50
C207 Interfacing Techniques	£7.50
Microprocessor Systems Design by Edwin Klingman ISBN 0-135-81413-8 (1977) 480 pages hard cover	16.40
CMOS Cookbook by Don Lancaster	£6.90
6800 Assembly Language Programming	£6.50
6800 Software Guide & Cookbook from Scelbi	£7.90
8080 Software Guide & Cookbook from Scelbi	£7.90
8080 A/8085 Assembly Language Programming	£6.50
8080A Bugbook Interfacing & Programming by Rony, Larsen & Titus (1977) 416 pages	£6.90
Z80 Programming for Logic Design by Adam Osborne	£5.90
Z80 Microcomputer Handbook by William Barden	£7.20
Structured Programming & Problem Solving with Pascal ISBN 0-138-54869-2 (1978) 365 pages by Kieburtz "When will your microcomputer speak Pascal"	£9.50
BASIC Computer Games — Microcomputer edition	£5.40
Best of BYTE Vol 1 (1977) 376 pages	£8.90
Best of CREATIVE COMPUTING Vol 1 (1977) 326 pages	£6.90
Vol 2 (1977) 323 pages	£6.90
MOI OTHELLO Game for 2 players on NASCOM 1 cassette	£2.00
MOI MINO PILOT powerful text editor and interpreter in ½KB!	£5.00

COMP COMPUTER COMPONENTS

(PART OF THE COMPSHOP LTD. GROUP)

For technical and sales literature send SAE to:
COMP COMPUTER COMPONENTS (Exidy UK Sales),
Freepost, 14 Station Road, New Barnet, Herts.
Tel: 01-441 2922 (Sales) 01-449 6596 Telex: 298755

TAKE ADVANTAGE OF OUR SHORT DELIVERY TIME.

All prices exclusive of VAT

NewBear

Computing Store Ltd.

New - Newbear DB 80 Printer £595

Serial and parallel ports
optional extras - 64 Key ASC11 Keyboard.
alternative character sets.

Introducing - Newbear 700

low cost VDU ex stock **£299**
V24 interface, upper case only keyboard and monitor available separately

Sym 1 6502 based micro from Synertek - fully assembled and tested
Down in price now only **£160**

Newbear now stocks O.K.Tools

Horizon (S100 bus Z80 based micros)

16K RAM with single disc drive **£1265**
32K RAM with double disc drive with 2 serial and 1 parallel port **£1983**

Schugart SA 400

5¼in disc drives down to **£190**

Now in stock BASF 5¼in disc drives £

Books

The Users Guide to North Star Basic R.Rogers £10.00 .75 pp

Basic with Business Applications Hayden £5.56 .50 pp

Basic Basic (An introduction to computer programming in Basic language) J.S.Coan £5.40 .50 pp

Practical Microcomputer Programming(6800) Weller £17.56 1.00 pp

Volume 0. The Beginners Book A.Osbourne £5.95 .50 pp

Volume 1.Basic Concepts A.Osbourne £5.95 .50 pp

Volume 11.Some Real Products A.Osbourne £18.95 1.00 pp

Volume 111.Some Real Support Devices A.Osbourne £11.95 1.00 pp

Understanding microcomputer and Small Computer Systems Scelbi £7.56 .50 pp

Book orders over £40 p&p free

Send for our complete book list

All prices subject to 8% VAT
Carriage by arrangement.
Barclaycard and Access welcome.
Minimum official order £10.

Callers welcome Mon - Sat 9.00 to 5.30 but please phone us first on 0635 30505

Visit our Northern office at
2A Gatley Rd.,
Cheadle, Cheshire.
Tel: 061-491 2290.

All Mail Order to Newbury.

24 HOUR DELIVERY IN MOST CASES

**Our new address is
Newbear Computing Ltd.
40 Bartholomew Street,
Newbury, Berks.**